Learning SQL Programming with Interactive Tools: From Integration to Personalization

PETER BRUSILOVSKY, SERGEY SOSNOVSKY, MICHAEL V. YUDELSON, DANIELLE H. LEE, VLADIMIR ZADOROZHNY, and XIN ZHOU
University of Pittsburgh

Rich, interactive eLearning tools receive a lot of attention nowadays from both practitioners and researchers. However, broader dissemination of these tools is hindered by the technical difficulties of their integration into existing platforms. This article explores the technical and conceptual problems of using several interactive educational tools in the context of a single course. It presents an integrated Exploratorium for database courses, an experimental platform, which provides personalized access to several types of interactive learning activities. Several classroom studies of the Exploratorium have demonstrated its value in both the integration of several tools and the provision of personalized access.

Categories and Subject Descriptors: K.3.1 [**Computing Milieux**]: Computers and Education—Computer Uses in Education–*Computer-assisted instruction (CAI)*; K.3.1 [**Computing Milieux**]: Computers and Education—Computer and Information Science Education–*Computer science education*

General Terms: Human Factors, Design, Experimentation

Additional Key Words and Phrases: Adaptive educational system, integrated learning environment, adaptive hypermedia, SQL

ACM Reference Format:
Brusilovsky, P., Sosnovsky, S., Yudelson, M. V., Lee, D. H., Zadorozhny, V. 2010. Learning SQL programming with interactive tools: From integration to personalization. ACM Trans. Comput. Educ. 9, 4, Article 19 (January 2010), 15 pages. DOI = 10.1145.1656255.1656257.
http://doi.acm.org/10.1145.1656255.1656257.

1. INTRODUCTION

Over the last 10 years, eLearning has emerged as one of the most popular types of Internet applications as well as one of the most active research

This material is based on work supported by the National Science Foundation under Grant #DUE-0633494.
Authors' addresses: P. Brusilovsky, S. Sosnovsky, M. V. Yudelson, D. H. Lee, V. Zadorozhny, and X. Zhou, School of Information Sciences, University of Pittsburgh, 135 N. Bellefield Ave. Pittsburgh, PA 15232; emails: peterb@pitt.edu, sas15@pitt.edu, mvy3@pitt.edu, suleehs@gmail.com, vladimir@sis.pitt.edu, reniorc@gmail.com.

areas. As practitioners argue for moving from passive and inefficient learning-by-reading to active learning-by-doing, many researchers have refocused their attention on innovative eLearning tools that support interactive and personalized learning [Berge 2002; Brusilovsky and Peylo 2003; Reeves 1999]. In a number of domains, one can already find a range of powerful, interactive, and (more rarely) personalized eLearning systems, varying from educational simulations to problem solving, some of which have wide international use. For example, math teachers offer interactive problems to their classes using WeBWorK [Hauk and Segalla 2005]. In the area of databases, thousands of students gain knowledge of Structured Query Language (SQL) by solving problems with the personalized SQL-Tutor [Mitrovic 2003], available to anyone who has purchased a database textbook from one of the major publishers.

Unfortunately, the use of innovative educational tools is not as broad as these tools deserve, due to the technical problems of integrating them into existing eLearning platforms such as Learning Management Systems (LMS). While simple educational content (HTML pages, slides) can be easily integrated by providing links to it or importing it as a part of a courseware package, rich interactive activities cannot be copied or referenced so easily. For example, SQL-Tutor problems are delivered and evaluated dynamically by a dedicated server where a student must log in before starting to work with a problem. After the problem evaluation is performed, the system has to store the results of the student's work. The teacher may want to monitor a student's progress. Students themselves will probably be interested to observe the improvement of their knowledge over time. The system needs to access this data in order to personalize its behavior to an individual student.

A recent analysis of existing LMS and other eLearning platforms [Rey-López et al. 2008] demonstrates that existing eLearning systems and standards do not provide necessary support for interactive or personalized learning content. As a result, LMS are predominantly filled with static, pedagogically inefficient content, while interactive content is available only through independent, self-contained systems such as WeBWorK [Hauk and Segalla 2005] and SQL-Tutor [Mitrovic 2003]. Most negatively, the existing situation affects students' ability to work with several kinds of interactive or personalized content within the same course. While a set of interactive tools can better support the needs of a particular course by complementing one another's strength, the technical difficulties of using multiple self-contained systems (each with its own login!) in the context of one course are too formidable for both teachers and students.

To use several advanced interactive systems in one course, one must resolve a number of problems, such as:

—how to allow students to access activities from different systems without multiple logins.

—how students' actions, observed by one system, can be stored in such a way that another system can utilize them.

—how the current knowledge of students can be derived from the logs of their activities in several different systems.

To investigate the technical problems and the pedagogical benefits of using several kinds of interactive learning tools, we have developed an integrated Exploratorium for database courses. The Exploratorium provides personalized access to three types of interactive learning activities: annotated examples, self-assessment questions, and a SQL lab. The architecture of the Exploratorium is open and the current version can be easily extended with additional activities and components. This article describes the components of the Exploratorium as well as its architecture, which allows integration of these components. It also presents the results of the system evaluation in six graduate and undergraduate database classes.

2. EXPLORATORIUM COMPONENTS

2.1 WebEx: Interactive Examples

The simplest kind of interactive content offered by the SQL Exploratorium is a large set of annotated SQL examples delivered by the WebEx system [Brusilovsky et al. 2004]. Each example consists of an SQL code fragment with an explanation for each important line. The WebEx interface allows students to interactively explore code explanations in arbitrary order (right part of Figure 1). By clicking on a checkbox, a student opens the explanation for the corresponding line of code. The click history is stored in the user model to produce an individualized, history-enriched environment for each student. Lines that have been explored by the student receive check marks. The color of the checkboxes indicates how many students in the class have accessed these lines of code (a more intense color indicates a higher number of students). These adaptive visual cues offer social navigation support [Dieberger et al. 2000], helping students to focus on the most important code fragments.

2.2 SQL-Knot: Knowledge Testing with Parameterized Problems

The SQL-KnoT (*Kno*wledge *T*ester) offers students an opportunity to evaluate and practice their problem-solving skills in the SQL domain. It generates questions that require a student to write an SQL query for a sample database, evaluates the correctness of the student's answer, and provides the student with feedback (left part of Figure 2). SQL-KnoT is similar to other Web-based testing systems [Kenny and Pahl 2005; Mitrovic 2003], but uses a novel approach to question generation and answer evaluation. Every time a student accesses an SQL-KnoT question, the actual question text is generated by the corresponding template from the set of predefined databases. When SQL-KnoT evaluates a student's answer, it randomly generates several starting states. After that, SQL-KnoT compares the result produced by the student solution for each database state with the corresponding result produced by the pre-stored correct query (model solution). To be evaluated as correct, the student solution must always produce the same result as the model solution. For the needs of our courses, we have developed about 50 templates capable of generating over 400 actual questions.

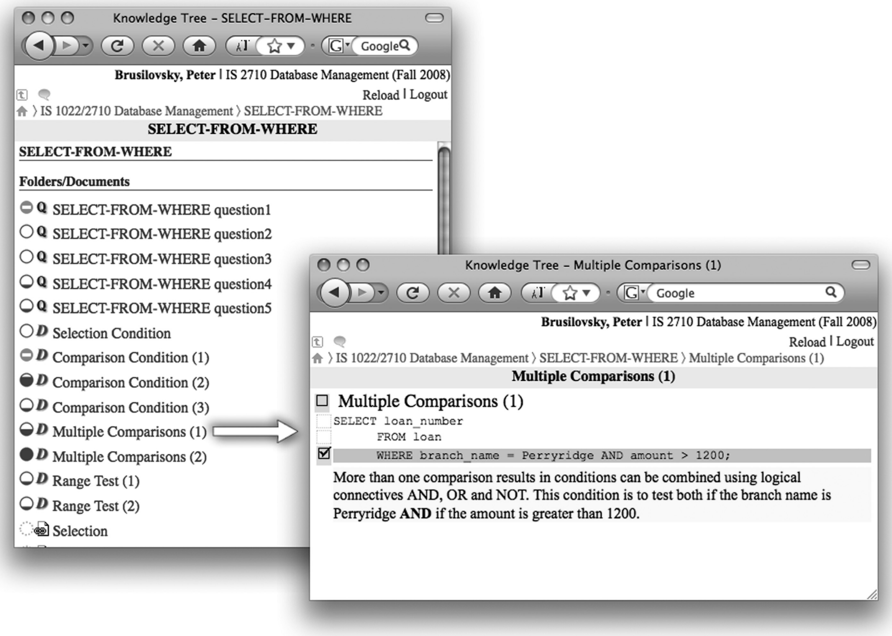

Fig. 1. A WebEx example accessed through the knowledge tree portal.

2.3 SQL-Lab: Focused Problem Exploration

SQL-Lab supports interactive exploration of SQL problem scenarios. The SQL-Lab interface allows students to formulate SQL queries and observe their results. The scripts can be designed by the students themselves, or copied from a set of WebEx examples. In addition, SQL-Lab can be accessed directly from SQL-KnoT questions in the problem-solving mode. In this mode, the student can see the question statement, work on the solution, and submit the tested solution to SQL-KnoT. Figure 2 shows a typical problem-solving scenario of using SQL-Lab. After failing to answer an SQL-KnoT question, a student opens SQL-Lab to run and debug his previous, erroneous solution.

3. INTEGRATION

As we mentioned in the introduction, there are no ready-for-use systems or architectures that can support integration of rich-content tools. The current, most popular, solution to integrate learning content from multiple sources is LMSs. Modern LMSs are built on a set of standards supporting integration and interoperability. However, as noted above, out analysis of several existing eLearning standards and LMSs has demonstrated that none of them can support integration of interactive or personalized learning content [Rey-López et al. 2008]. As a result, the authors of innovative interactive content systems such as WeBWorK [Hauk and Segalla 2005], SQL-Tutor [Mitrovic 2003], BOSS [Joy et al. 2005], CourseMarker [Higgins et al. 2005], and similar tools

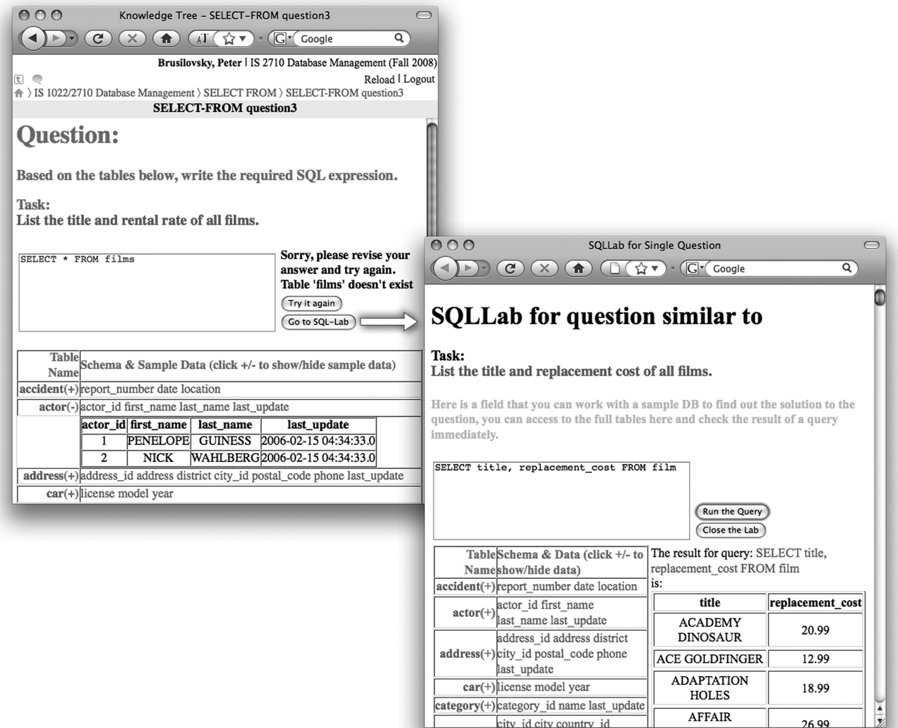

Fig. 2. A problem presentation in SQL-KnoTwith SQL-lab opened to develop and debug the answer.

[Douce et al. 2005] have to offer their rich content through specialized portals with dedicated components for registering users and logging their interactions. This approach supports interactivity and personalization, but does not support integration.

In this situation, researchers and practitioners who want to support integrated access to interactive and personalized content from multiple providers have to develop various *integration architectures* such as APeLS [Conlan and Wade 2004], XTA [Nuzzo-Jones et al. 2005], or MEDEA [Trella et al. 2005]. This is the direction taken by our team as well. The Exploratorium is based on our own integration architecture ADAPT^2 (Advanced Distributed Architecture for Personalized Teaching and Training) [Brusilovsky et al. 2004]. ADAPT^2 attempts to both integrate rich interactive content and support personalization.

The ADAPT^2 approach to system integration is based on recognizing two main goals. First is to achieve the technical integration, that is, to have diverse interactive content accessible from one Web portal with a single sign-on. Moreover, the portal should not dictate how teachers structure this content, but instead allow them to structure the content according to their preferred way of teaching. The second goal is to keep a history of students' interactions with each component accumulated in a standard format in an accessible centralized repository. This approach makes system components aware of each other and opens the door to implementing various kinds of personalization. We refer to it as conceptual integration.

Within ADAPT^2, the technical integration of interactive content is supported by the Knowledge Tree portal. For students, it offers centralized access to all three kinds of learning content: WebEx interactive examples, SQL-KnoT problems, and SQL-Lab. For teachers, the portal serves as an LMS. It allows teachers to structure the learning content according to the needs of their course. The interface of Knowledge Tree is based on a common folder-document paradigm. Each course is structured with a sequence of nested folders separating the course material. For example, if a teacher chooses to structure the material lecture-wise, the folders will represent lectures and contain individual resources (SQL-KnoT problems, WebEx examples, etc.) relevant to particular lectures. An example of such a folder aggregating the learning material for the lecture on "SELECT-FROM-WHERE" queries is shown in the left part of Figure 1.

The conceptual integration in ADAPT^2 is supported by the user modeling server, CUMULATE [Yudelson et al. 2007]. CUMULATE accepts reports about students' activity from all of the educational tools: navigational clicks from WebEx interactive examples, problem-solving attempts in SQL-KnoT, and the application of problem-related skills in SQL-Lab. These reports are used by CUMULATE to infer the current level of students' knowledge for each concept of the SQL domain, which can later be requested by adaptive system components for various kinds of personalization. For example, WebEx uses CUMULATE to visualize individual and group click traffic (Figure 1). QuizGuide (introduced below) uses student knowledge models to annotate links to SQL-KnoT problems with navigational cues. Knowledge Tree portal implements several adaptive navigation techniques to help students choose the best learning activity. The two-way information exchange between interactive learning content and CUMULATE is performed using standard HTTP-based protocols.

We hope that the work on ADAPT^2 and similar integration architectures will influence the development of modern LMSs and eLearning standards in such a way that the LMS of the future will be able to directly integrate interactive personalized content [Nuzzo-Jones et al. 2005]. Meanwhile, the Exploratorium could be used in parallel with (or instead of) a traditional LMS. Modern single sign-on mechanisms can be used to reduce the burden of logging into several systems.

4. PERSONALIZATION

A range of possible techniques can be applied to make an eLearning system personalized [Brusilovsky and Peylo 2003]. In our case, the challenge has been to select the most appropriate personalization approach for an integrated system, which can work with a variety of interactive learning content. For this project, we have chosen a *topic-based adaptive navigation support* approach. In the past, we have explored this approach in the QuizGuide system, which has

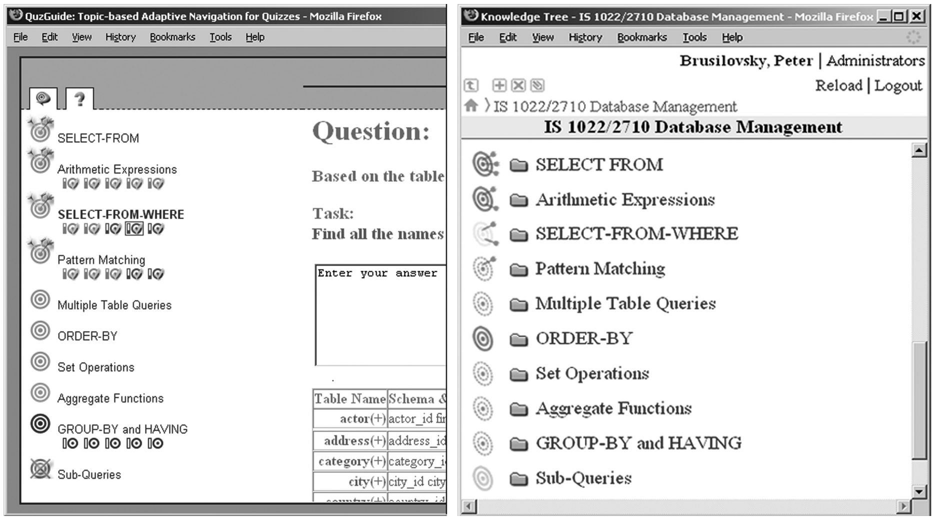

Fig. 3. Topic-based navigation support in QuizGuide (left) and the knowledge adaptive portal (right).

been successfully used for helping students study C programming [Brusilovsky and Sosnovsky 2005]. Our studies demonstrated that this approach is both easy to understand and very efficient: it increased both the quality of student work and their commitment to work with the system [Brusilovsky and Sosnovsky 2005]. In the first version of the Exploratorium, we used a modified version of the QuizGuide system to personalize student access to SQL-KnoT problems [Sosnovsky et al. 2008]. After the effectiveness of this approach in database classes was confirmed, we expanded the scope of topic-based navigation support to all interactive content by developing an adaptive version of the Knowledge Tree portal. Both versions of this approach are presented below.

4.1 QuizGuide: Adaptive Navigation for SQL Questions

QuizGuide provides an alternative personalized way to access SQL-KnoT questions. In QuizGuide, questions are grouped into 20 topics. To guide the student adaptively to the most appropriate topic to practice, QuizGuide annotates the link to each topic with an adaptive “target with arrows” icon (Figure 3, left). The icon presentation is based on two factors:

—the individual student’s performance on that topic (the number of arrows in the target increases as the student demonstrates progress within the topic’s questions);

—the relevance of the topic to the current goal of the course (bright-blue target color designates the current topic, light-blue is the prerequisite for a current topic, gray means that the topic is not-relevant to the current learning goal, while a crossed-out target means that the student is not ready for the topic yet).

4.2 An Adaptive Course Portal

Capitalizing on our experience with QuizGuide, we expanded the topic-based navigation support approach to all kinds of learning content by merging the personalization power of QuizGuide with the portal's ability to provide access to all kinds of content. A topic-based personalization service was embedded into the original Knowledge Tree portal. For each folder of the course, the service retrieves from CUMULATE current levels of users' progress measured for all learning activities aggregated by the folder, and generates personalized navigation support icons. For the icon, we reused the target-arrows paradigm used in QuizGuide. The number of arrows indicates student knowledge growth through successful problem solving. No arrows denote no or little success with problems in the folder; 1, 2, or 3 arrows denote low, medium, and high success (Figure 3, right). The color of the target indicates student progress in exploring supporting content (examples, animations, etc.). If the student has not explored any example in the folder, the target is grey; otherwise, the target is shown in a green color of differing intensities: the darker it looks, the more exploration the student has done.

5. CLASSROOM STUDIES

Several large-scale classroom studies have been performed to investigate different aspects of Exploratorium usage. Overall, the Exploratorium has been introduced to 229 students in six graduate and undergraduate database courses:

—two undergraduate and one graduate courses at the University of Pittsburgh;

—two undergraduate courses at the National College of Ireland;

—one undergraduate course at the Dublin City University, Ireland.

In these studies, we adjusted various settings such as the presence of a particular kind of adaptation for a particular type of interactive learning content (see Table I) or reinforcing students with extra credits for heavy usage of the system (Pitt-F'07-U in Table I). However, the general set of Exploratorium tools (interactive examples, SQL-Knot problems, and SQL-Lab) was available for all courses, and the number of elementary activities within these tools stayed the same. Overall, the students explored 24,753 lines of interactive examples, made 10,963 attempts to answer SQL-Knot problems, and performed 4,031 exploratory actions with SQL-Lab. Table I summarizes the basic statistics of tool usage.

Overall, 75% of the students tried the Exploratorium tools at least once. On average, more than 7 out of 10 students who tried the Exploratorium tools continued to work with them. This is a fairly high retention ratio indicating a high level of satisfaction on the part of the students. An average student answered about 73 of the SQL-KnoT problems, explored 155 lines of examples, and used the SQL-Lab 29 times over the duration of the course. The use of the tool was not mandatory and the students received no extra credit for their work with Exploratorium (with the exception of the Pitt-F'07-U course).This volume of

Table I. Cumulative Statistics of Exploratorium Usage

Course	**Personalization**	**Number of students**		**Mean number of interactions**		
		Registered	**Tried the tools**	**SQL-KnoT**	**SQL-Lab**	**WebEx**
Pitt-F'07-U*	QuizGuide	38	27	156.44	10.84	209.12
Pitt-F'07-G	QuizGuide	37	24	61.65	13.37	95.54
Pitt-S'08-U	QuizGuide	33	24	28.81	26.38	54.89
NCI-S'08-U1	None	25	17	12.71	2.17	25.13
NCI-S'08-U2	None	23	18	7.89	2.50	33.94
DCU-S'08-U	Portal	73	62	90.13	54.96	250.31

* F'07 – Fall 2007 semester, S'08 – Spring 2008 semester, U – undergraduate, G – graduate.

activity confirms that the students considered the tools to be useful for learning. The subjective evaluation based on the post-course questionnaires confirms this data (see Section 5.3).

The Exploratorium was introduced to students as a collection of tools for self-assessment of knowledge and preparation for homework assignments and tests. The statistical evaluation shows that there is a significant correlation between students' performance in class and the amount of work they completed within Exploratorium. Students in the undergraduate courses (Pitt-F'07-U and Pitt-S'08-U) who explored more WebEx examples also received better grades on their homework assignments (Kendall's tau correlation coefficient = 0.19; $p = 0.032$). A similar relation has been observed between the amount of students' work with SQL-KnoT and their homework grades (Kendall's tau correlation coefficient = 0.22; $p = 0.012$).

5.1 The Importance of Integration

Three main Exploratorium systems allow students to master SQL skills on different cognitive levels. For example, a student who has trouble with a SQL problem can learn from similar examples or practice queries in the SQL-Lab. The evaluation results show that students appreciated the presence of several different tools. Although some students used only a single tool, the majority of them (about 84%) worked with all systems in a balanced manner. Even in a single session, they often switched between different systems to explore the variety of content available for their topic of study. Out of a total of 882 sessions, less than half were dedicated to a single system. The majority of sessions had student interactions with at least two of the main Exploratorium tools. In more than a quarter of the sessions, students used all three systems. Figure 4 visualizes distribution of registered sessions among the Exploratorium tools. The statistical analysis of all sessions averaged by students shows that the number of sessions involving more than one tool ($M = 2.19$, $SD = 1.65$) is significantly higher than the number of sessions where a student worked with one system in isolation ($M = 1.74$, $SD = 1.79$) Wilcoxon Signed-Ranks Z-statistic = -2.613, $p = 0.009$. This data confirms the importance of the integration of multiple interactive educational activities in a single system, which is one of the core ideas of the Exploratorium.

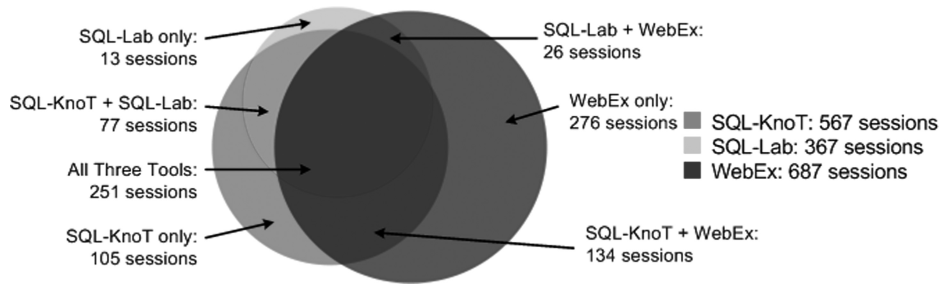

Fig. 4. Session distribution.

5.2 The Value of Personalization

Personalization plays an important role in increasing the usability of the content served by the Exploratorium. The major personalization technology employed is adaptive navigation support, which is known for bringing several benefits to eLearning. Among the benefits are the promotion of nonsequential navigation to the most appropriate learning content [Brusilovsky and Eklund 1998], improvement of learning outcomes [Davidovic et al. 2003; Specht 1998], higher motivation to work with educational activities [Brusilovsky et al. 2009b; Weber and Brusilovsky 2001], and increase in the speed of learning [Brusilovsky and Pesin 1998; Masthoff 2002]. Our experience with the topic-based adaptive annotation implemented in QuizGuide has confirmed most of these effects.

To investigate the motivational value of adaptive navigation support, we compared the number of times the SQL-KnoT problems were accessed, the number of distinct problems attempted, and the number of sessions in adaptive and non-adaptive modes. In three courses at the University of Pittsburgh, the same set of SQL-KnoT problems was available to students in the adaptive manner through QuizGuide system (see Figure 3) and through the non-adaptive portal (see Figure 1). Overall, in these courses, students made 4,404 attempts to solve problems via QuizGuide and 1,514 attempts to solve problems via the portal in the non-adaptive mode. As the data show (see Table II-a) the students were much more willing to access problems in the adaptive mode (through QuizGuide) and use of the adaptive mode caused them to work more with the system. On average, students made almost three times as many attempts in the adaptive mode than they did in the non-adaptive. They also accessed more distinct problems when receiving adaptive navigation support from QuizGuide. The difference in the amount of work done might be caused by more frequent access (a greater number of sessions) and/or by longer sessions. For both of these parameters, we observed higher values in the adaptive mode than in the non-adaptive. This agrees with our previous findings [Brusilovsky and Sosnovsky 2005] that in the presence of adaptive navigation support students not only access the system more frequently, but also stay with the system longer and do more work per session. Courses offered at NCI and DCU were different from the ones offered at University of Pittsburgh: NCI students had

Table II. Comparison of Cumulative Usage Parameters for Adaptive and Non-Adaptive Access to SQL Problems

a) Within-subject tests (Pitt-F'07-U, Pitt-F'07-G, Pitt-S'08-U)			
	Adaptive	**Non-Adaptive**	**Significance**
Mean # of problem attempts	113.00	38.44	$p = .015$, $Z = -2.19$
Mean # of distinct problems attempted	24.13	15.31	$p = .019$, $Z = -2.08$
Mean # of sessions	3.43	2.57	$p = .044$, $Z = -1.71$

b) Between-subject tests (NCI-S'08-U1, NCI-S'08-U2, DCU-S'08-U)			
	Adaptive	**Non-Adaptive**	**Significance**
Mean # of problem attempts	90.13	10.23	$p < .001$, $Z = -6.10$
Mean # of distinct problems attempted	24.81	5.26	$p < .001$, $Z = -6.20$
Mean # of sessions	4.65	2.14	$p < .001$, $Z = -5.10$

only the non-adaptive interface, and DCU students had only the adaptive interface. Nevertheless, between-subject comparison also shows that those who work with the adaptive interface answer more problems in general and explore wider variety of problems (see Table II-b).

To confirm the effect of adaptation on the promotion of non-sequential navigation through the learning content, we performed an analysis of the navigational patterns followed by the students using QuizGuide. The analysis shows that when a student decides to switch from one topic to another, an average of half of these transitions did not lead to the next topic in the list. Those students chose to jump to a different topic following the annotations served by QuizGuide, guiding them to what is currently more appropriate content. This observation is interesting outside of the context of our work. It provides evidence that freedom of navigation is important for students and that eLearning systems may not use the best strategy by forcing all students to follow a predefined module-to-module path.

5.3 Student Feedback Analysis

A questionnaire was administered at the end of the semester in four undergraduate courses using the Exploratorium at the University of Pittsburgh and NCI. The questionnaire collected students' opinions about a number of different issues using a Likert scale from 1 (strongly disagree) to 5 represented (strongly agree). This section analyzes students' answers for eight questions that are the most relevant to the focus of this article (see Figure 5). Since we observed no significant difference between the answers of students in different classes, we integrated data from all classes. To ensure informed feedback, we excluded answers from students who had not completed a sufficient amount of activity with the Exploratorium. In total, the following analysis integrates answers from 52 students.

The results of the questionnaire demonstrate a very positive attitude towards all the Exploratorium tools. For instance, 91% of the students agree or strongly agree that the use of annotated examples helped them during the course (AE1) and 67% thought similarly about SQL-Lab (SL). 90% of the SQL-KnoT users agree or strongly agree that the problems in SQL-KnoT were intellectually challenging (SK2). The personalization offered through navigation

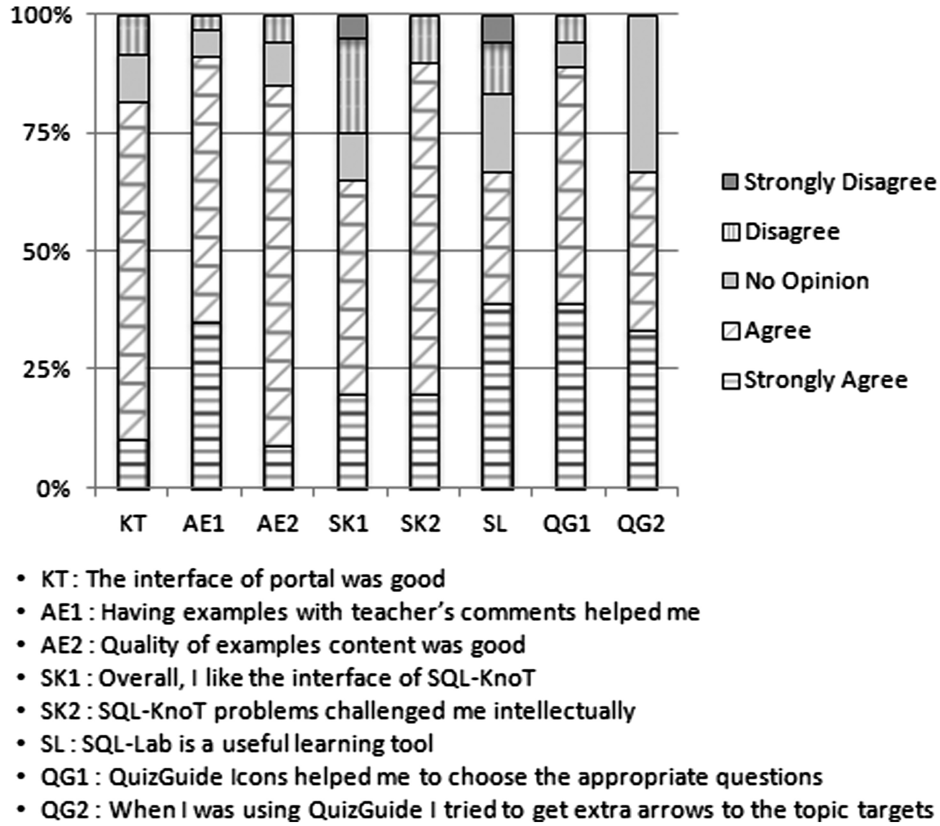

Fig. 5. Results of the student evaluation of the Exploratorium.

support was highly appreciated, as well: 89% of the students agree or strongly agree that QuizGuide personalized topic icons helped them to choose the appropriate questions (QG1).

In addition to component-focused questions, we asked students to rank the main system components by their value to the course (1 = the most important; 5 = the least important). Student answers to this question provided more evidence in favor of both the integrated and the personalized nature of the system. Overall, different students championed different tools, showing that none of the tools was a silver bullet satisfying everybody. As a result, the average component rank varied just between 2.70 and 3.68. The key personalized component, QuizGuide, got the best ranking of 2.70 with the next, SQL-KnoT trailing at 3.28.

After closer examination, we have also found that the amount of work with annotated examples positively and reliably correlates with their perceived usefulness (Kendall's tau statistic = .325, p = .013). In other words, the more often annotated examples were used, the better the student's attitude towards them. This, once again, stresses the role of personalized navigation support, which encourages students to use interactive content and thus to appreciate its value.

6. SIMILAR WORK

Two streams of research are directly relevant to our projects: 1) interactive eLearning tools for SQL and databases, and 2) personalized, integrated eLearning systems. At the moment, both areas are under-explored. While many interactive learning tools have been created to teach computer science subjects, only a few are focused on databases and SQL. These systems can be roughly classified into two groups. The first group supports students learning through interactive examples, demonstrating the basic concepts of SQL and illustrating their use in practice [Guimaraes 2006; Pahl et al. 2004]. These examples are often created based on multimedia technology. The tools from the second group support learning-by-doing: offering students SQL problems and evaluating their solutions [Kenny and Pahl 2005; Mitrovic 2003]. Both approaches have been proven effective. For example, after using the SQL-Tutor system, students have demonstrated a significant improvement in problem solving performance on the post-test [Mitrovic 2003].

Our tools (SQL-KnoT, WebEx, and SQL–Lab) are similar to these efforts, while providing several specific innovations. For example, WebEx offers a new kind of content (annotated examples) for the SQL learning domain. SQL-KnoT explores parameterized question generation and an original approach to evaluation of students' solutions.

Architectures for the personalized integration of several interactive eLearning systems have been explored by just a handful of projects such as MEDEA [Trella et al. 2005] and APeLS [Dagger et al. 2003]. Driven by the same motivation as ADAPT^2, these architectures explored different technical solutions, allowing the community to learn a number of valuable lessons. Currently the developers of these systems are collaborating in an attempt to design cross-framework, personalized integration approaches, which will hopefully lead to commonly accepted integration standards for interactive systems.

7. SUMMARY AND FUTURE WORK

We presented a personalized Exploratorium for database courses. This system provides access to several types of advanced educational activities, each served by an independent Web-based tool. The Exploratorium integrates diverse interactive content, while allowing the teacher to structure access to this content according to the needs of his course. The system tracks students' work with all of its components and builds a user model for every student, which is used to adapt some of the components based on individual progress. The Exploratorium has been evaluated in six graduate and undergraduate database courses in three different universities in two countries. The evaluation demonstrates that both the integration and personalization features of the system are important. The students extensively used all components of the system. The provision of personalization further increased their motivation to work with it. The students' feedback about the system has been highly positive.

In our future work, we plan to advance the current state of the system in several directions. First, we want to expand the Exploratorium as both a personalized and an integrated system. We are currently developing several cross-content personalization approaches, such as recommendation of relevant examples or readings after an unsuccessful attempt to solve an SQL-KnoT problem. We are also working on integrating additional educational tools into the Exploratorium including both new components from our team and systems developed by other researchers. The most recent version of the Exploratorium already integrates the multimedia examples presented in Pahl et al. [2004]. With the support of NSF, we are now working on integrating SQL-Tutor [Mitrovic 2003] as another personalized component of the Exploratorium. Some early results of this work are reported in Brusilovsky et al. [2009a]. We welcome the opportunity to collaborate with other developers of educational tools for the SQL domain.

Our second goal is to make the Exploratorium broadly available to students and teachers of Database courses. Although the current stage of our work is mostly focused on pedagogical and architectural issues, we are trying to collaborate with instructors outside of our home university who wish to pilot the Exploratorium in their courses. We would like to further extend this collaboration and encourage instructors who are interested in using our system to contact us. We expect that a broader use of the Exploratorium will help us to gradually refocus our work from pedagogical issues to scalability problems, and eventually turn the Exploratorium into a truly open community system.

REFERENCES

BERGE, Z. L. 2002. Active, interactive, and reflective eLearning. *Q. Rev. Dist. Educ. 3*, 2, 181–190.

BRUSILOVSKY, P. 2004. KnowledgeTree: A distributed architecture for adaptive e-learning. In *Proceedings of the 13th International World Wide Web Conference (WWW'04)*. 104–113.

BRUSILOVSKY, P. AND EKLUND, J. 1998. A study of user-model based link annotation in educational hypermedia. *J. Univ. Comput. Sci. 4*, 4, 429–448.

BRUSILOVSKY, P., MITROVIC, A., SOSNOVSKY, S., MATHEWS, M., YUDELSON, M., LEE, D., ZADOROZHNY, V. 2009. Database exploratorium: A semantically integrated adaptive educational system. In *Proceedings of the Ubiquitous User Modeling Workshop at the 17th International Conference on User Modeling, Adaptation, and Personalization (UMAP'09)*.

BRUSILOVSKY, P. AND PESIN, L. 1998. Adaptive navigation support in educational hypermedia: An evaluation of the ISIS-Tutor. *J. Comput. Inform. Techn. 6*, 1, 27–38.

BRUSILOVSKY, P. AND PEYLO, C. 2003. Adaptive and intelligent Web-based educational systems. *Int. J. AI Educ. 13*, 2-4, 159–172.

BRUSILOVSKY, P. AND SOSNOVSKY, S. 2005. Engaging students to work with self-assessment questions: A study of two approaches. In *Proceedings of the 10th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE'05)*. 251–255.

BRUSILOVSKY, P., SOSNOVSKY, S., AND YUDELSON, M. 2009b. Addictive links: The motivational value of adaptive link annotation, *New Rev. Hypermedia Multimedia 15*, 1, 97–118.

BRUSILOVSKY, P., YUDELSON, M., AND SOSNOVSKY, S. 2004. An adaptive e-learning service for accessing Interactive examples. In *Proceedings of the World Conference on E-Learning (E-Learn'04)*. 2556–2561.

CONLAN, O. AND WADE, V. P. 2004. Evaluation of APeLS: An adaptive eLearning service based on multi-model, metadata-driven approach. In *Proceedings of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'04)*. 291–295.

DAGGER, D., CONLAN, O., AND WADE, V. P. 2003. An architecture for candidacy in adaptive eLearning systems to facilitate the reuse of learning resources. In *Proceedings of the World Conference on E-Learning (E-Learn'03)*. 49–56.

DAVIDOVIC, A., WARREN, J., AND TRICHINA, E. 2003. Learning benefits of structural example-based adaptive tutoring systems. *IEEE Trans. Educ. 46*, 2, 241–251.

DIEBERGER, A., DOURISH, P., HÖÖK, K., RESNICK, P., AND WEXELBLAT, A. 2000. Social navigation: Techniques for building more usable systems. *Interactions 7*, 6, 36–45.

DOUCE, C., LIVINGSTONE, D., AND ORWELL, J. 2005. Automatic test-based assessment of programming: A review. *ACM J. Educ. Res. Comput. 5*, 3, Article 4.

GUIMARAES, M. 2006. The Kennesaw Database Courseware (KDC): Strong points, weak points, and experience using it in a classroom environment. *J. Comput. Sci. Coll. 21*, 3, 91–96.

HAUK, S. AND SEGALLA, A. 2005. Student perceptions of the Web-based homework program WeBWorK in moderate enrollment college algebra classes. *J. Comput. Math. Sci. Teach. 24*, 3, 229–253.

HIGGINS, C., GRAY, G., SYMEONIDIS, P., AND TSINTSIFAS, A. 2005. Automated assessment and experiences of teaching programming. *ACM J. Educ. Res. Comput. 5*, 3, Article 5.

JOY, M., GRIFFITHS, N., AND BOYATT, R. 2005. The BOSS online submission and assessment system. *ACM J. Educ. Res. Comput. 5*, 3, Article 2.

KENNY, C. AND PAHL, C. 2005. Automated tutoring for a database skills training environments. In *Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'05)*. 58–62.

MASTHOFF, J. 2002. Design and evaluation of a navigation agent with a mixed locus of control. In *Proceedings of the 6th International Conference on Intelligent Tutoring Systems (ITS'02)*. 982–991.

MITROVIC, A. 2003. An intelligent SQL tutor on the Web. *Int. J. AI Educ. 13*, 2-4, 173–197.

NUZZO-JONES, G., WALONOSKI, J., HEFFERNAN, N., AND LIVAK, T. 2005. The eXtensible tutor architecture: A new foundation for ITS. In *Proceedings of the Workshop on Adaptive Systems for Web-based Education at 12th International Conference on Artificial Intelligence in Education (AIED'05)*. 1–7.

PAHL, C., BARRETT, R., AND KENNY, C. 2004. Supporting active database learning and training through interactive multimedia. In *Proceedings of the 9th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE'04)*. 27–31.

REEVES, T. C. 1999. A research agenda for interactive learning in the new millennium. In *Proceedings of the 11th World Conference on Educational Multimedia, Hypermedia and Telecommunications (EMHT'99)*. 15–20.

REY-LÓPEZ, M., BRUSILOVSKY, P., MECCAWY, M., DÍAZ-REDONDO, R. P., FERNÁNDEZ-VILAS, A., AND ASHMAN, H. 2008. Resolving the problem of intelligent learning content in learning management systems. *IJEL 7*, 3, 363–381.

SOSNOVSKY, S., BRUSILOVSKY, P., LEE, D. H., ZADOROZHNY, V., AND ZHOU, X. 2008. Reassessing the value of adaptive navigation support in e-learning. In *Proceedings of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'08)*. 193–203.

SPECHT, M. 1998. Empirical evaluation of adaptive annotation in hypermedia. In *Proceedings of the 10th World Conference on Educational Multimedia and Hypermedia and World Conference on Educational Telecommunications (EDMEDIA/EDTELECOM'98)*. 1327–1332.

TRELLA, M., CARMONA, C., AND CONEJO, R. 2005. MEDEA: An open service-based learning platform for developing intelligent educational systems for the Web. In *Proceedings of the Workshop on Adaptive Systems for Web-based Education at 12th International Conference on Artificial Intelligence in Education (AIED'05)*. 27–34.

WEBER, G. AND BRUSILOVSKY, P. 2001. ELM-ART: An adaptive versatile system for Web-based instruction. *Int. J. AI Educ. 12*, 4, 351–384.

YUDELSON, M., BRUSILOVSKY, P., AND ZADOROZHNY, V. 2007. A user modeling server for contemporary adaptive hypermedia: An evaluation of push approach to evidence propagation. In *Proceedings of the 11th International Conference on User Modeling (UM'07)*. 27–36.

Received January 2009; revised July 2009, August 2009; accepted September 2009