ExplaNet: A Framework to Manage and Analyze Student-Authored Course Content

Jessica Masters[*], *Tara M. Madhyastha*[†], *Ali Shakouri*[‡]

1 Introduction

It has become clear that the Web should be a useful tool for education. The ability to self-publish is a major driving force behind this. Instructors can easily share course content with students and other instructors. In addition, on-line quizzes, applets, and simulations can allow students to take a more active role in the learning process. Unfortunately, the problem of creating course modules that meet the needs of a broad audience is analogous to the reuse of software modules: it is difficult to create modules that can be widely reused. This process is expensive, and with constantly emerging technological areas, modules need to be continually updated.

We have created a framework, ExplaNet, that allows students to author course content for each other in a guided assessment environment. ExplaNet can be seen as an extension to course discussion boards, which are frequently used to post and answer student questions with instructor moderation. In ExplaNet, the instructors post questions for students to solve, posting their explanations. Explanations can be submitted in a variety of multi-media formats. If questions cover important concepts, this will generate a lot of course content. Student who do not understand the questions will be able to access the explanations provided by other students. Students can also rank explanations, creating an individual profile for each student that can be used to customize the explanations provided in the future. Instructors can use ExplaNet to assess student answers and trace the flow of information throughout the class.

The remainder of this paper is organized as follows. Section 2 describes related work in the area of on-line quizzing and assessment and collaborative filtering. Section 3 describes ExplaNet in detail. Next, in Section 4, we describe preliminary experiments with ExplaNet and the results. We conclude in Section 5 with a summary of our work thus far, and the directions for future work.

2 Related Work

There have been several models developed to categorize how students take in and process information [3, 5, 6, 9]. By using these models, instructors can create courseware that appeals to students with a variety of learning preferences. There is currently a wealth of such course materials. In the area of Electrical Engineering alone, several on-line applets and simulations have been designed to help students

[*]Department of Computer Science, University of California at Santa Cruz, jmasters@soe.ucsc.edu
[†]Department of Computer Engineering, University of California at Santa Cruz, tara@soe.ucsc.edu
[‡]Department of Electrical Engineering, University of California at Santa Cruz, ali@soe.ucsc.edu

learn in an active web environment. The State University of New York at Buffalo department of Electrical Engineering has created multiple web-based Java applets in areas including crystal structure, metal-oxide semiconductors, digital circuits, and semiconductor devices [20]. The Visual Quantum Mechanics project at Kansas State University has developed on-line materials in quantum physics for both the high school and college level [21]. The University of Maryland offered a new course in quantum physics that stressed the important of students becoming actively involved in the learning process [11]. These are only some of the resources available, and only in one field of engineering. Despite the large volume of existing active learning materials, the development of these materials is costly, both in time and resources.

In addition to these materials crafted for a specific topic, there have been several educational software packages designed to facilitate on-line testing and quizzes. WebCT [19] and Blackboard [1] are two existing courseware management software packages that include the functionality of on-line newsgroups as well as instructor-created on-line quizzing. There have also been many research efforts in this domain. QuizIT, an on-line quiz software, was designed to improve the Computer Science department at Virginia Tech by making the grading and evaluation process efficient, while not increasing the instructor time required to create quizzes [18]. PILOT was designed specifically to test graphing problems in computer science [2]. These tools, however, usually create traditionally structured quizzes: students take quizzes on their own, submit their answers, and receive some form of feedback. ExplaNet attempts to blend several ideas, allowing for the creation of customized course content, while providing tools for student assessment without high costs to the instructor.

One idea central to ExplaNet is that we can mine student-authored explanations and filter explanations to reduce the amount of irrelevant or useless information each student is exposed to. Collaborative filtering systems use historical preferences to predict future preferences. The key assumption is that if student A is "like" student B in some meaningful way, student A's explanation will be more useful to student B than for some very different student C. Student rankings are obtained explicitly, by ranking icons provided in ExplaNet, as well as implicitly. An implicit but positive ranking, for example, would be logged if a student downloaded and saved another student's explanation to their own computer.

Collaborative filtering has been implemented to sort content ranging from movies to news articles. Tapestry, the earliest system to implement collaborative filtering, was designed to filter e-mail documents by having users write "annotations" for their documents [4]. GroupLens is an extensive research project at the University of Minnesota that applies collaborative filtering techniques to Usenet news articles to predict which articles users will want to read [7, 8, 12, 14]. PHOAKS (People Helping One Another to Know Stuff) uses collaborative filtering to sort recommendations of websites that have been posted in Usenet news articles [17]. Siteseer is a system that uses collaborative filtering to predict websites for users based on other users bookmarking these pages[13].

3 ExplaNet

ExplaNet is a web-based software tool that instructors can use to post questions on-line. Students upload files containing multi-media explanatory answers to these questions. The system can be customized to allow any type of file, ranging from a simple text file to a picture or a word processing document. ExplaNet then presents other students' explanations and allows students to rank those explanations. ExplaNet is web-based and consists of Java Servlets [16] that access a MySQL database [10].

We believe an instructor is more likely to expend effort on materials for assessment than on tailored course materials. Furthermore, while an instructor can only provide one point of view, each student brings to the class a different background, perspective, and unique experiences. ExplaNet exploits this diversity by shifting the process of creating course content from the instructor to the student. This will create a pool of moderated content, thus increasing the likelihood of there being an explanation in the pool to fit the learning needs of each student.

Students can benefit by seeing various explanations of concepts, provided in a variety of formats. In addition, by filtering the explanations that are presented to each student, we can reduce the volume of extraneous or poorly suited information provided to each student. This filtering can be based both on the history of rankings made by each student, learning preferences or general information about the student. Instructors can use the tool for assessment by treating it like traditional quiz software. In addition, instructors can use ExplaNet to identify concepts that have been widely misunderstood, as well as to determine what students consider to be a correct or well-defined answer. Instructors can also see trends in behavior and trace the path of information, or misinformation.

3.1 The Menu Frame

Figure 1 is a screen shot of ExplaNet. On the left of the screen, Figure 1(a), students are presented with a collapsible menu containing all quizzes, questions, and student-authored explanations to date. This menu was designed to look familiar, similar to a newsgroup. When a student signs in, all quiz and question folders are closed, as "Quiz 2: C Basics" is in Figure 1(a). When a student clicks on a quiz to open it, it expands to show the questions contained, like "Quiz 1: Unix Basics." The questions are also presented as closed folders, with only the first few characters of the question appearing, until a student opens that question. When a student clicks on a question folder to open it, a list of answers appears.

The answer list is sorted according to an answer's rank. We use the student's explicit rank, unless the student has not yet ranked the answer, in which case we estimate the rank. Answers of the same rank are randomized, so as not show a preferences for certain answers. The rank is listed as either "My Rank" or "Estimated Rank." The creator of the answers is listed with an anonymous name such as "anonUser27." Figure 1(a) shows that this user has submitted an answer to the question, and three other students have submitted their answers as well. The answer created by "anonUser4" has an estimated rank of five, while the answer created by "anonUser2" has been given an explicit rank of four by the user. The currently viewed question is highlighted in red, and the currently viewed answer has a red arrow pointing to it, like the answer created by "anonUser2" in Figure 1(a), which is the answer currently being viewed. Each answer is actually a link; when clicked, the link affects the view in the other frames.

An instructor can use ExplaNet with a variety of options. One option is that students can sign in and immediately see all answers submitted to date. This option is useful, for example, if the question is meant to foster discussion on a topic. Another option is that students can see other answers only after they have submitted their own answer. This option can be used for more quiz-like questions. Finally, time constraints can be used. Students have until the first deadline to submit their answer, and only after that answer is submitted and the deadline has passed, can they see other answers and submit more answers. This is useful if an instructor wants the majority of answers to be submitted before allowing other students to view them.

3.2 The Submit Frame

Figure 1(b) shows the Submit Frame, on the top right hand side of the screen. This frame prints the currently open quiz and question and shows the student how many answers they have currently submitted

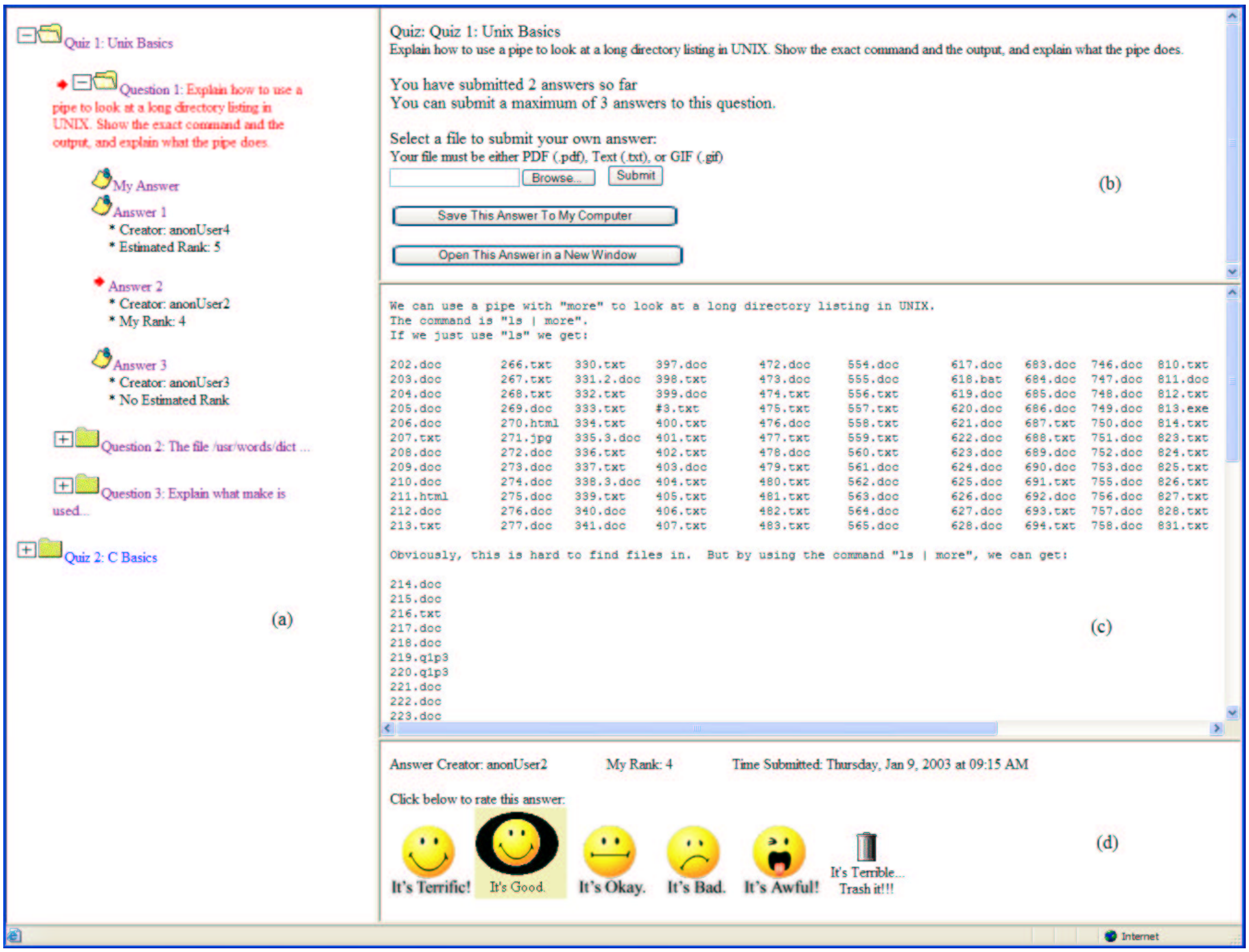

Figure 1: Screen Shot of ExplaNet. (a) The Menu Frame presents the list of quizzes, questions and submitted answers. (b) The Submit Frame allows the student to upload a file as their answer to a question. (c) The Answer Frame displays the currently selected answer. (d) The Ratings Frame displays the possible ranks a student can give an answer, as well as the current rank the student has given.

to this question. Figure 1(b) shows that this student has submitted two answers to this question, and could submit one more answer if he or she wishes. An instructor can allow students to submit any number of answers. Students upload the file in the Submit Frame. If a question is multiple choice, this frame also contains the choices, one of which must be selected in addition to uploading the answer file containing the explanation of their choice. Instructors can either allow the wrong answer choice to be submitted, or reject all incorrect multiple choice answers.

Also in this frame are two buttons, as seen in Figure 1(b). One opens the answer in a new window. This is useful if the answer is particularly large. Another button allows the user to download and save the answer to their personal computer. This is useful if a student finds an answer extremely helpful and wants to refer to it often, without having to sign back in to the ExplaNet framework.

3.3 The Answer Frame

In the middle of the right hand side of the screen is a large frame, as seen in Figure 1(c). The actual answer is displayed in this frame. If an application is needed to view an answer (e.g. Microsoft Word, Adobe Acrobat), this application will open in this window. In the future, we hope to transform this frame into an editing window, so students can author their explanations directly into the Answer Frame, rather than independently editing the answer and uploading it.

3.4 The Ratings Frame

At the bottom of the right hand side of the screen is the ratings frame, as seen in Figure 1(d). Here there are five smiley-face icons. They correspond to the possible ratings a user can have for an answer. The ratings range from 1–5. Once an answer is ranked, that rank is highlighted. Figure 1(d) shows that the answer currently being viewed has been given a rank of four. There is also a trash can icon. If the students "throws away" an answer, it is rated with a -1, and will never appear on their list of answers again. If the student is looking at their own answer, this frame also tells the student how many times their answer has been viewed by other students. This is meant to encourage students and make them aware that other people truly are reviewing their work.

If an instructor is using ExplaNet, there is one more icon in this frame. This icon allows an instructor to "destroy" an answer. Unlike throwing an answer away, where the answer is only removed from an individual's answer list, this option allows an instructor to remove the answer from circulation. If an answer is particularly bad or misleading, the instructor can destroy it, and it will never appear on any student's answer list. Also, the creator of the answer is sent an e-mail telling them their answer was rejected, and they must submit a new answer. This can be helpful in stopping the spread of misinformation.

4 Evaluation

We tested ExplaNet in EE70, "Introduction to Electronics", in the Electrical Engineering department at the University of California at Santa Cruz, Fall 2002. This is a required class for Computer Engineering and Electrical Engineering majors. In the Fall of 2002 there were approximately 60 students in the class. Introduction to Electronics is a core class that introduces main electrical engineering concepts, including the physical basis and mathematical models of electrical components and circuits, and an introduction to circuit and network design. It is a prerequisite for more advanced engineering courses related to analog

Submissions	Question 1	Experiment 1 (a) Question 2	Combined	Question 1	Experiment 2 (b) Question 2	Combined
Number of Students Submitting Answers	49	51	100	48	45	93
Number of Answers Submitted	58	70	128	59	58	117
Percentage of Students Who Submitted a Second Answer	18.37%	37.25%	28.00%	22.90%	28.90%	25.81%
Views						
Total Number of Views by Students	209	208	417	301	401	702
Average Number of Answers Viewed by Each Student	5.92	4.08	4.17	6.30	8.90	7.55
Average Number of Answers Viewed by Students who Submitted a Second Answer	4.67	2.95	3.59	14.30	24.50	19.83
Ranks						
Total Number of Ranks by Students	122	82	204	84	44	128
Average Number of Answers Ranked by Each Student	2.49	1.61	2.04	1.80	1.00	1.38
Average Rank Students Gave Answers (-1, or 1–5)	3.48	3.84	3.62	3.20	2.80	3.06
Number of Ranks Students Gave Their Own Answers	17	21	38	10	15	25
Average Rank Students Gave Their Own Answers	4.35	4.05	4.18	4.00	3.80	3.88
Saves						
Total Number of Answers Downloaded and Saved	0	0	0	1	2	3

Table 1: Student usage results from two ExplaNet quizzes given to EE70 at UCSC in Fall, 2002.

circuits, signals and systems, communications, etc. We performed two experiments, giving two quizzes to the students. ExplaNet records all student activity: submission, viewing, and ranking of answers. We analyzed the usage patterns and activity during the experiment.

4.1 ExplaNet Student Usage

Each quiz contained two questions. For both quizzes, students could submit at most two answers. Students could only see other answers after their first answer was submitted. For this experiment, there was no collaborative filtering of answers. Answers were presented in a purely random order to the students. There were no estimated ranks.

4.1.1 Experiment 1: Transient Behavior in Circuits

There were two questions on the first quiz, designed to test if students knew how transient change in current or voltage in circuits with resistors and capacitors can be calculated. Students were asked to use mathematical descriptions of circuits to solve problems in everyday applications. Finally, students were to

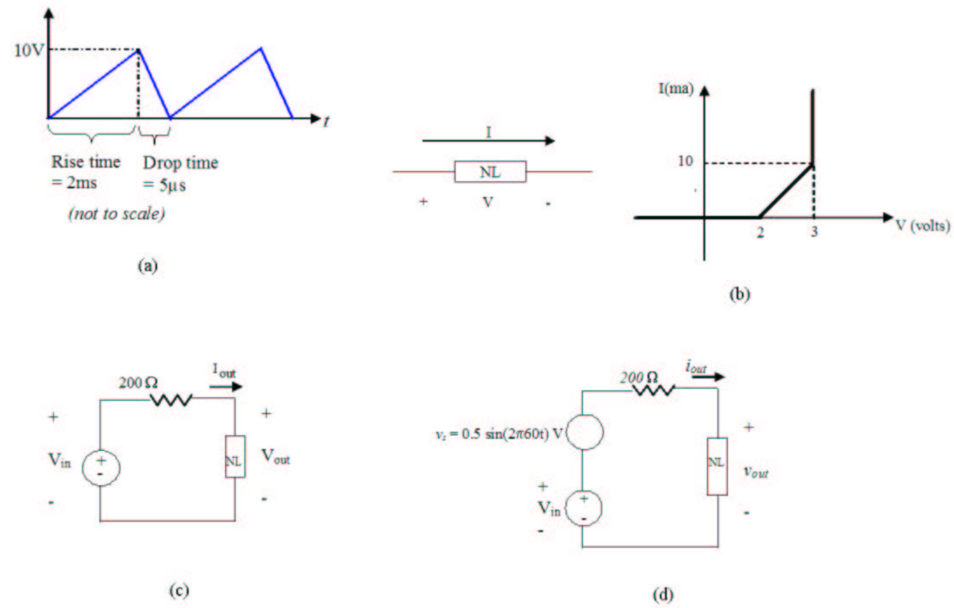

Figure 2: Figures included in the ExplaNet quizzes Given to EE70, Fall 2002.

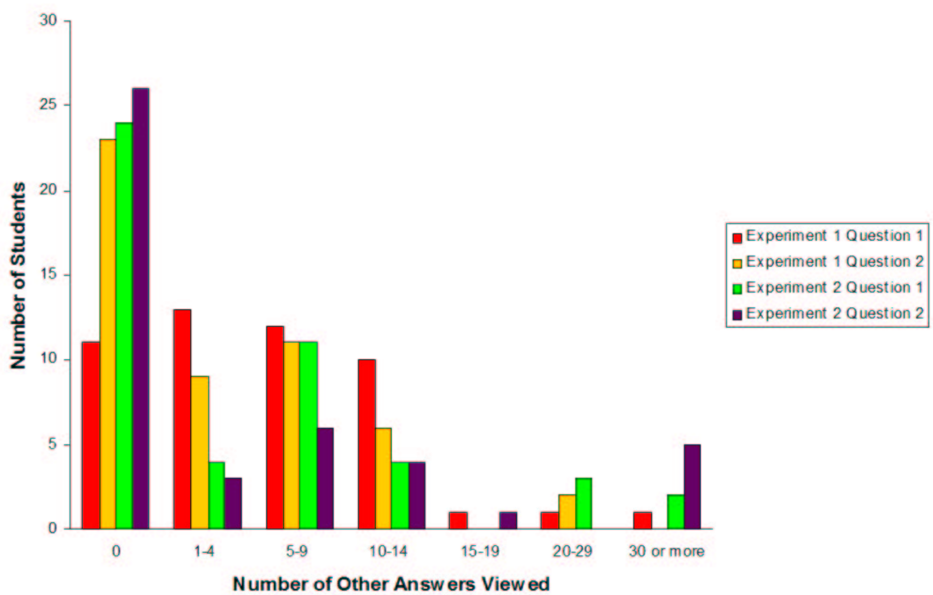

Figure 3: Number of answers viewed by each student.

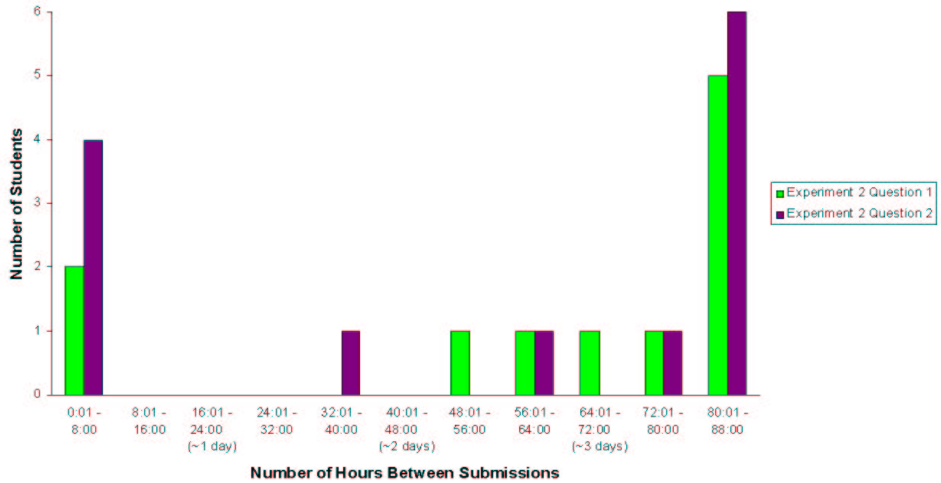

Figure 4: Time in hours between the first and second answer submission.

calculate the derivative of a piecewise linear function using graphical technique.

Question 1 was "When a capacitor is connected to a DC source, its voltage rises from 20V to 36V in 4μ with an average charging current of 0.6A. Determine the value of the capacitance." Question 2 was "To move the spot of a cathode-ray tube across the screen requires a linear increase in the voltage across the deflection plates, as shown here in Figure 2(a). Given that the capacitance of the plates is 4nF, sketch the current flowing through the plates." Table 1(a) shows the usage results for the first experiment. It includes data about the number of answer submissions, the viewing of students' answers, the ranking of students' answers, and the downloading of students' answers.

The data from this first quiz shows us that students view, on average, four answers. Even for those students taking the time to re-think their answer, and submit a second answer, there were, on average, under four answers viewed in between the first and second submissions. It is extremely important, therefore, to choose these few answers very carefully. If answers are selected randomly, it is not likely that students will find an answer that appeals to them within the first five answers. However, if we can craft a sophisticated algorithm for choosing the answers, hopefully the first few answers presented to the student will be well-suited to explain the question.

Figure 3 shows the number of answers viewed by each student. We can see there were a large number of students, 22.4% for Question 1, and 45.0% for Question 2, who viewed zero answers. Many educators have discovered that no matter how much effort is invested into creating an educational tool, some students will not take advantage of these tools. This evidence, unfortunately, confirms this belief.

Table 1(a) also shows that students ranked even fewer answers than they viewed: on average 1.38. This leads a common problem in collaborative filtering, known as the sparsity problem [12]. To reduce the rating sparsity, we plan to use information relating to student learning preferences. In other words, where ranks cannot be correlated between users to estimate a rank for a new answer, we will correlate their learning preference information. We can also see from Table 1 that students consistently ranked their own answers higher than answers created by other students.

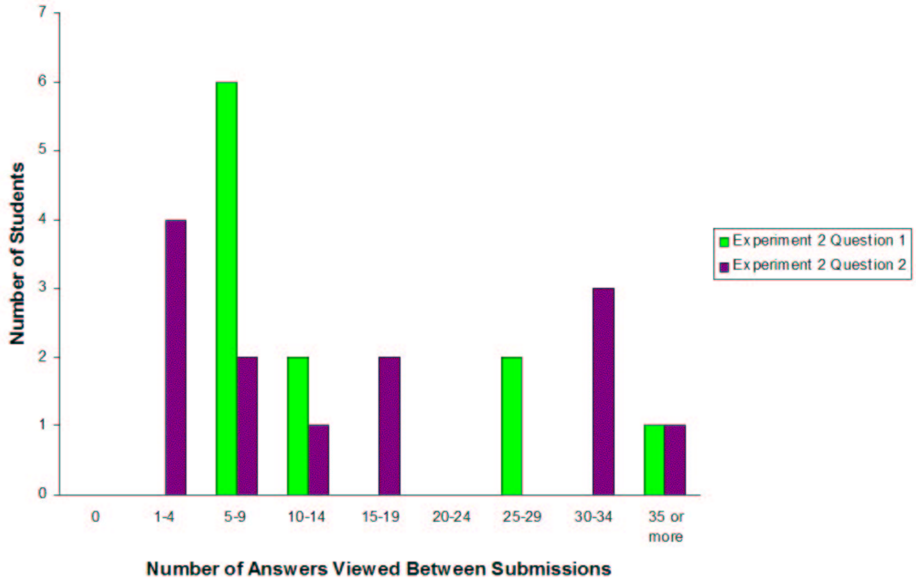

Figure 5: Number of answers viewed between the first and second answer submission.

4.1.2 Experiment 2: Nonlinear Circuit Element, Load Line Analysis Technique

One problem we discovered with the first experiment was that the first few answers submitted had the most number of views and ranks. This is because these answers were available to other students the longest. But the first few answers submitted are not necessarily the best answers or the best answers for any given individual. To correct this problem, we structured the second experiment differently. We used the idea discussed in Section 3.1 of one deadline for the first submission. After that deadline, students can view all other answers. Now, a majority of answers would be submitted to the system before choosing which answers to present to each student. In addition to this change, we offered a loosely worded "bonus" for students who viewed and ranked several answers.

The second quiz was designed to designed to test the techniques of nonlinear circuit element and load line analysis techniques. First, the quiz asked whether or not students could calculate the current and voltage in a circuit that has a non-linear element. Next, it tested whether students could use a load line technique to obtain the DC bias condition. The quiz asked if students knew how the non-linear element can be replaced by an equivalent linear element for small signals. Finally, the quiz asked if students could analyze that the equivalent linear element depends on the DC bias condition.

Question 1 was "Here is the current-voltage characteristic of a new non-linear circuit element, Figure 2(b). What are the output voltage and current in the DC circuit in Figure 2(c)? Three cases: (i) V_{in} =1V, (ii) V_{in}=2.2V, (iii) V_{in}=4V." Question 2 was "We added a small signal sinusoidal source at the input, Figure 2(d), plot output voltage versus time. Three cases: (i) V_{in}, (ii)V_{in}2.2V, (iii)V_{in}=4V (in each case what is the small signal equivalent circuit of the non-linear element)."

Table 1(b) shows the student usage results for these questions. Even with our encouragement to view and rank more answers, students still viewed only a relatively small number of answers: 7.55 views on average. Figure 3 shows the number of answers viewed. Even more students did not view any answers for this experiment, 50% for Question 1, and 58% for Question 2. For this experiment, students who

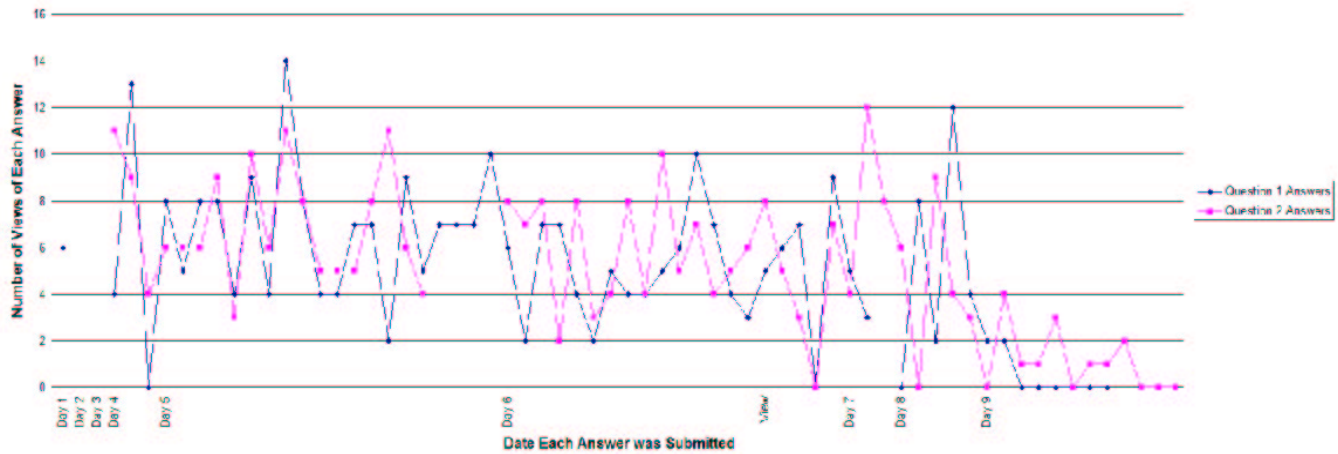

Figure 6: The number of times answers submitted for the second experiment were viewed.

submitted a second answer did view many more answers on average than those who did not, but this is only one small step in the right direction, since only 28% of students submitted a second answer.

Also despite our encouragement, there were fewer ranks for each student in this experiment, with each student ranking 1.38 answers on average, as seen in Table 1(b). For this experiment, students finally utilized the ability to download and save other students' answers. In addition to using learning preference information to account for the sparsity of ranks, we can also use the downloading and saving of an answer as an implicit high rank for that answer.

Figure 4 shows the time between the first and second submission of answers. From this we can see that for those students who submitted two answers, most waited at least two days between their first and second submission. Figure 5 shows the number of answers viewed between the first and second submission. While most students waited at least two days between submits, they did not necessarily view many answers in that time period. In fact, there is no correlation (Pearson coefficient $r = 0.155$) between the time spent between submits and the number of answers viewed during that time. The number of answers viewed between submits is a much better indicator of activity than the time between submits.

4.2 ExplaNet Answer Activity

We analyzed the answer activity for the second experiment. Because of the structure of the first experiment, the activity for each answer was biased by when that answer was submitted. Therefore, we only present the activity for the second experiment.

Figure 6 shows the number of times each submitted answer was viewed. Answers are represented as points, plotted according to when they were submitted. The "View" label indicates that at this time other answers were available for students to view, and students could also then submit a second answer. There is a fairly even distribution of views for each answer, at least those submitted before the time at which answers could be viewed. This makes sense because the answers were presented randomly. This is a large improvement over the first experiment, where the distribution was highly skewed towards answers submitted early on. Using this information, we concluded that using ExplaNet with the structure defined for this second experiment is much more beneficial.

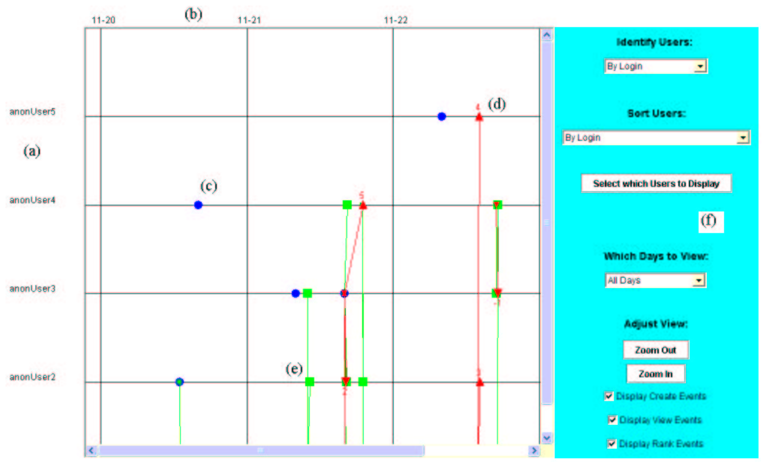

Figure 7: ExplaNet timing graphs are used to trace student activity. (a) Students in the class. (b) Days the quiz was available for students to take. (c) Round blue dot: a new answer has been submitted by this student at this time. (d) Red triangle: a student has ranked another student's answer at this time. (e) Green square: a student has viewed another student's answer at this time. (f) Display options to change the view of the time graph.

Our hypothesis, however, is that a small number of "quality" answers will rise to the top of ExplaNet. Clearly, this cannot happen without filtering of the answers. With random answer selection, all answers have an equally likely chance of being viewed, as seen in Figure 6. To consider student opinions, we clearly need a filtering algorithm. Hopefully, this algorithm will find the best answers submitted, and those answers will be seen by a maximum number of users.

4.3 ExplaNet Timing Graphs

We created a tool for analysis that is now built into ExplaNet. At any time, it is useful for an instructor to be able to see the progress of a quiz, or to be able to look at some of the answers that have been submitted. When an instructor signs into ExplaNet, there is an icon in the Menu Frame to display the time graph for each question. We included this tool for instructors to be able to keep frequent tabs on student usage and activity.

4.3.1 The ExplaNet Time Graph Applet

We created a Java Applet [16] that displays a time graph much like the type of graphs used for train schedules. Figure 7 is a screen shot of this time graph with fictional data to simplify the display. The time graph is a visual representation student activity. Each horizontal line represents a student. Lines are connected with diagonal vertical lines when a student views or ranks another answer. The right-hand side of the screen, Figure 7(f), shows the panel of display options available to the instructor, explained below. A time graph can be drawn for each individual question.

The students are listed on the left-hand side of the graph, as seen in Figure 7(a), with a horizontal line for each student. The student names can be clicked and a pop-up window will appear with more information about the student, as can be seen in Figure 8(d). Instructors can use the options "Identity Users," "Sort Users," and "Select Which Users to Display" in the options panel, Figure 7(f), to change

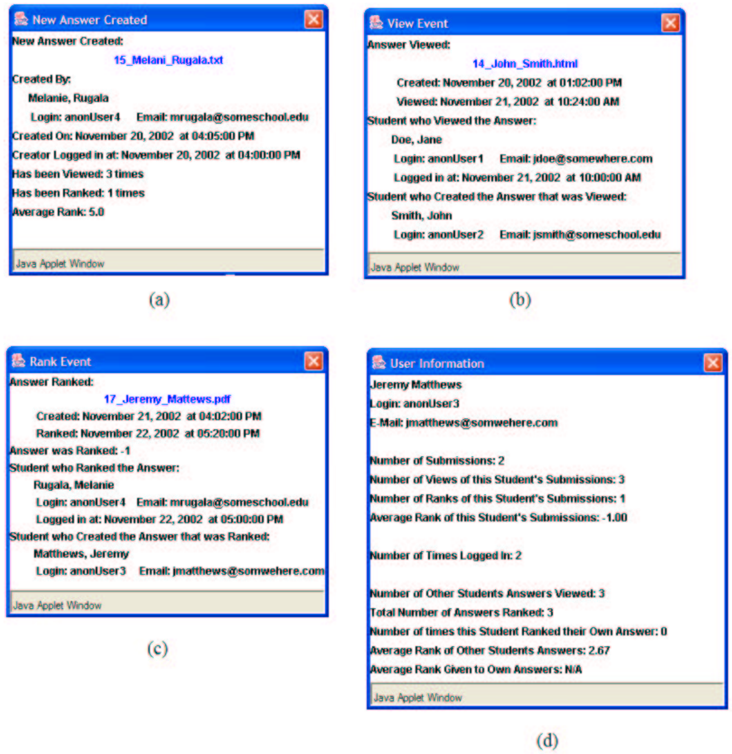

Figure 8: ExplaNet timing graph pop-up windows. (a) Extra information displayed about the submission of a new answer. (b) Extra information displayed about the viewing of an answer. (c) Extra information displayed about the ranking of an answer. (d) Extra information displayed about a student.

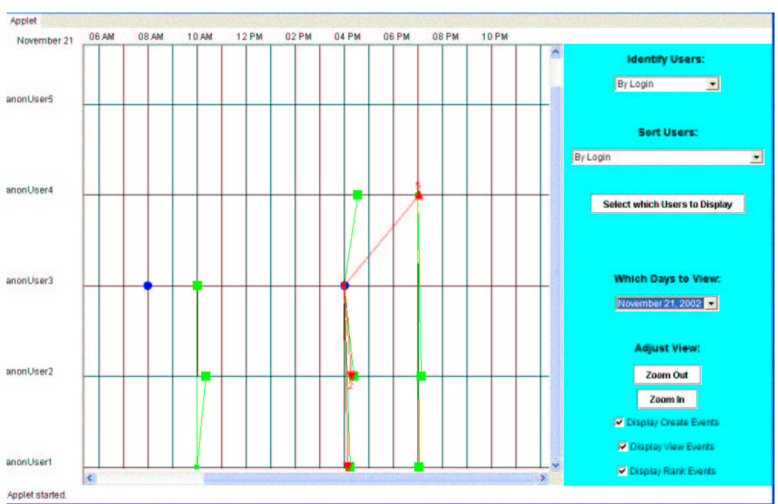

Figure 9: ExplaNet timing graphs: one day, more detailed view with vertical lines now corresponding to a specific hour of the selected day.

the appearance of the list of students. Instructors can identify students by either their name, anonymous login, or e-mail address. By clicking "Select Which Users to Display," a list of all users pops up, and the instructor can select or de-select any student individually. This allows the instructor to look at a specific subset of the class.

Instructors can sort by student names, logins, or student activity. Students can be sorted according to who has viewed the most answers, for instance, if an instructor wants to see which students have been the most active during a quiz. Students can also be sorted according to who created the answer that was viewed the most times. Instructors can sort students according to who ranked the most answers, or by who created the answer thats was ranked the most. Finally, students can be sorted by who created the highest ranked answer, or by who, on average, ranked other answers the highest. Using these sorting options, at any time during the quiz, an instructor can sort by which answers have been viewed the most, and then only look at the top two or three answers to see what information is being spread throughout the class. An instructor could also sort by the highest ranked answers, and look at the top two or three answers to get a general idea of what students think is a good or quality answer. This can be especially useful for instructors with very large classes, where looking at every submitted answer is not a reality.

The dates of the quiz are listed on the top of the graph, Figure 7(b). The vertical lines correspond to these days, creating a time line. Instructors can choose to view the entire duration of the quiz, as seen in Figure 7, or can use the "Which Days to View" option in Figure 7(f) to select a specific day to be viewed. Figure 9 shows the time graph with only one day of activity being viewed. Now the labels across the top refer to the time of day. Instructors can use this view to get a more detailed account of the activity. For either the day or complete view, instructors can use the zooming buttons in Figure 7(f) to zoom in or out of the graph and get a more or less detailed view.

The round blue dots, Figure 7(c), refer to the submission of a new answer. The answer dot represents an answer submitted by the student whose horizontal line the dot appears on. The x-axis position of the dot represents the time the answer was submitted. An instructor can click on this dot to get more information about this particular answer. Figure 8(a) is the pop-up screen that appears when an instructor clicks on a round submission dot. All pop-up windows in Figure 8 contain a hyperlink to the answer itself, which will open in a new window.

Green lines with square endpoints represent one student viewing another student's answer, as seen in Figure 7(e). The student with the small green square is viewing the answer of the student with the large green square. The originating x-axis position of the line corresponds to when the student logged into ExplaNet. The ending x-axis position of the line corresponds to when the viewing of the answer occurred. This explains the slight diagonal slant of the lines. The squares can also be clicked to display more information about the viewing event, as shown in Figure 8(b).

Red lines with triangle endpoints represent one student ranking another student's answer, as seen in Figure 7(d). The lines are slanted like the viewing lines. The larger red triangle, corresponding to the answer being ranked, also displays the actual rank given. Red triangles can also be clicked for more information about the rank, as seen in Figure 8(c).

4.3.2 Time Graphs Data from Experiments

Figure 10 is the timing graph for Experiment 2, Question 1. Figure 11 is the timing graph for Experiment 2, Question 2. Note that these quizzes were given before the final version of the timing graph software was complete, and we did not have the login information for the students. Therefore we have estimated the starting position for viewing and ranking lines. We can see from the graphs that the majority of answers

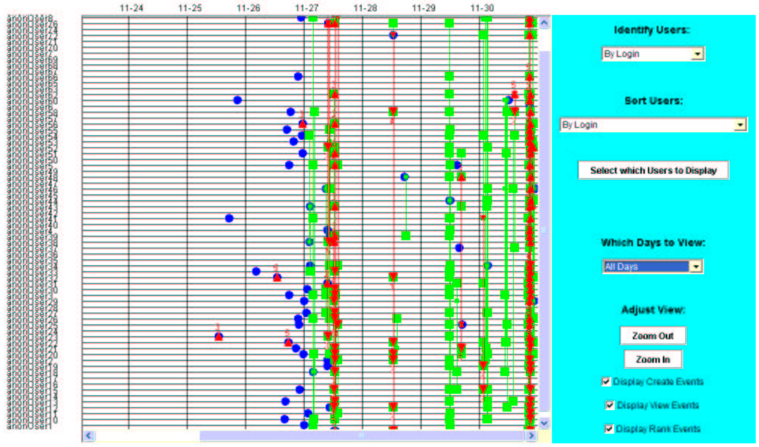

Figure 10: ExplaNet time graph, EE70, UCSC Fall 2002, quiz 2, question 1.

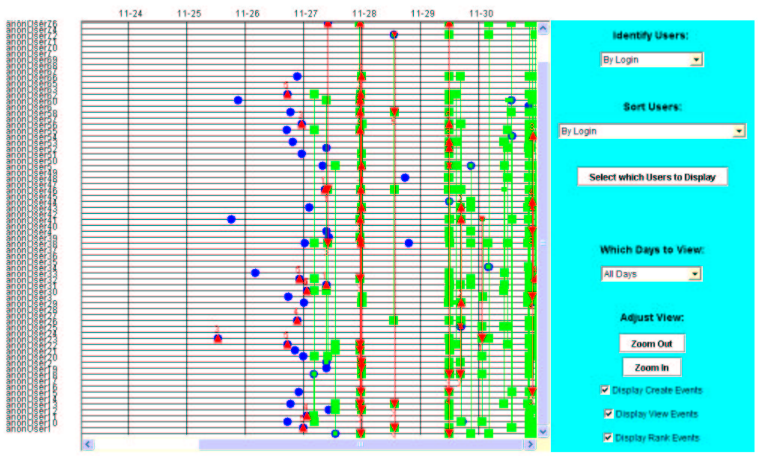

Figure 11: ExplaNet time graph, EE70, UCSC Fall 2002, quiz 2, question 2.

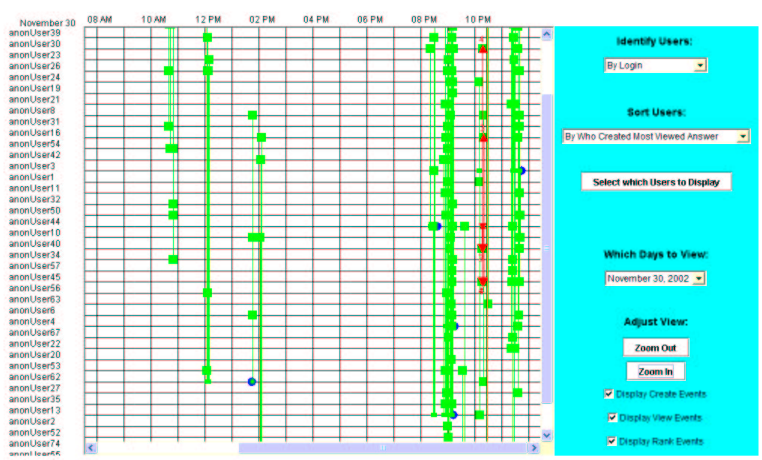

Figure 12: ExplaNet time graph, EE70, UCSC Fall 2002, quiz 2, question 2, one-day view for November 30.

were submitted on November 27. November 27 at 10:00 am was the first deadline for submission and the first time that other students' answers were allowed to be viewed. Very few answers were submitted after this date, which indicates very few re-submits. Next, we select only to view November 30; the final submissions were allowed up until midnight on this day. Also, we can sort by whose answers were viewed the most and zoom in on these students. Figure 12 shows the applet with these changes for Question 2. It is interesting to see the large amount of activity that takes place in the last remaining hours before the quiz submission closes. Also, this time period is only days before the final exam. This could explain the cluster of activity as students use explanations to prepare for finals.

5 Conclusions and Future Work

We created ExplaNet as a tool for both students and instructors. Students can benefit by having access to many explanations, authored by other students, each with a different point of view and method of explanation. We included tools that allow instructors to monitor student activity. Furthermore, the evaluation tools allow instructors to select only a few answers, giving an overview of student thought patterns.

From our first experiment with ExplaNet, we were able to characterize student usage patterns. With no algorithm for choosing which answers to display to users, all submitted answers had an even distribution. This is not ideal. Our hypothesis is that a small subgroup of answers are of high quality, and that there are only a few answers suited to each individual. Our experiment showed that students will only look at a handful of answers, so those answers need to be chosen carefully. In addition, even though viewing and ranking an answer is as simple as clicking a link, many students will not use this tool. Perhaps we cannot fix this problem, but because of how collaborative filtering works, this can contribute to the sparsity problem which will affect students who do wish to utilize the tool. Therefore, we need other student information to use in choosing answers and correlating users.

Our first experiment with ExplaNet has lead us to make some changes in the framework. We will be running a large experiment with the new framework at the University of California at Santa Cruz in Winter, 2003, using CMPS 12B, "Introduction to Data Structures". This class is a required class for Computer Science, Computer Engineering, and Electrical Engineering majors, and there are approximately 130 students in the class. The first change is that the rank of a student's answer will affect their grade, via extra credit. This will hopefully encourage students to create more thoughtful answers. To prevent students from precariously ranking their friend's answers high, each user will have a unique mapping of other users to pseudonyms. For example, if student A logs in to ExplaNet, every answer created by student B will be listed as being created by "anonUser27." However, when student C logs in, every answer created by student B will be listed as being created by "anonUser43." This way each user has their own mapping of pseudonyms, and student B cannot tell his or her friends "rank me high, my pseudonym is anonUser13." Students will not be allowed to rank their own answers.

Another important change is that the menu will list the top three "featured answers." We will take the highest rated answers to date, and randomly choose three of them to be "featured." The featured answers will appear first in the list, with a special icon to distinguish them. Finally, we will implement the first version of our collaborative filtering. It will be very simple: if student A has ranked two of student B's answers in the past, any new answers by student B will have the estimated ranking of the average of the past rankings. This is obviously very primitive, but we feel it is sufficient for our first pass. In addition, students will take the Felder-Soloman Index of Learning Styles [15] so we can look for correlations between rankings and learning preferences.

Acknowledgments

We would like to thank Leslie Clark for her work and efforts in analyzing the data from our experiments.

This material is based upon work supported by the National Science Foundation under Grant No. CCR 0093051 and Grant No. CCL1 0088881.

References

[1] Blackboard. Welcome to Blackboard. http://www.blackboard.com. Retrieved December, 2002.

[2] Bridgeman, Goodrich, Kobourov, and Tamassia. PILOT: An interactive tool for learning and grading. *SIGCSEB: SIGCSE Bulletin (ACM Special Interest Group on Computer Science Education)*, 32, 2000.

[3] R.M. Felder and L.K. Silverman. Learning and teaching styles in engineering education. *Engineering Education*, 78(7):674–681, 1988.

[4] David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 25(12):61–70, 1992.

[5] Ned Herrmann. *The Creative Brain*. Brain Books, 1990.

[6] David Kolb. *Experiential Learning: Experience as the Source of Learning and Development*. Prentice-Hall, 1984.

[7] Joseph A. Konstan, Bradley N. Miller, David Maltz, Jonathan L. Herlocker, Lee R. Gordon, and John Riedl. GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87, March 1997.

[8] Bradley N. Miller, John T. Riedl, and Joseph A. Konstan. Experience with GroupLens: Making Usenet useful again. In USENIX, editor, *1997 Annual Technical Conference, January 6–10, 1997. Anaheim, CA*, pages 219–233, Berkeley, CA, USA, 1997. USENIX.

[9] Isabel Briggs Myers. *Gifts Differing*. Consulting Psychologists Press, 1980.

[10] MySQL. MySQL: Speed, power and precision. http://www.mysql.com. Retrieved December, 2002.

[11] E. F. Redish, R. N. Steinberg, and M. C. Wittmann. A new model course in applied physics. http://www.physics.umd.edu/perg/qm/qmcourse/NewModel/, 2001. Retrieved December, 2002.

[12] P. Resnick, N. Iacovou, M. Suchak, P. Bergstorm, and J. Riedl. GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pages 175–186, Chapel Hill, North Carolina, 1994. ACM.

[13] James Rucker and Marcos J. Polanco. Siteseer: Personalized navigation for the web. *Communication of the ACM*, 40(3):73–75, March 1997.

[14] Badrul M. Sarwar, Joseph A. Konstan, Al Borchers, Jonathan L. Herlocker, Bradley N. Miller, and John Riedl. Using filtering agents to improve prediction quality in the GroupLens research collaborative filtering system. In *Computer Supported Cooperative Work*, pages 345–354, 1998.

[15] Barbara A. Soloman and Richard Felder. Index of learning styles questionnaire. http://www2.ncsu.edu/unity/lockers/ users/f/felder/public/ILSdir/ilsweb.html. Retrieved December, 2002.

[16] Sun Microsystems, Inc. The source for Java technology. http://java.sun.com. Retrieved December, 2002.

[17] Loren Terveen, Will Hill, Brian Amento, David McDonald, and Josh Creter. PHOAKS: A system for sharing recommendations. *Communications of the ACM*, 40(3):59–62, March 1997.

[18] L. Tinoco, D. Barnette, and E. Fox. Online evaluation in WWW based courseware. In *Proceedings of the Twenty-eighth SIGCSE Technical Symposium on Computer Science Education*, pages 194–198, February 1997.

[19] WebCT. WebCT. http://www.webct.com. Retrieved December, 2002.

[20] C. R. Wie. The semiconductor applet service. http://jas2.eng.buffalo.edu/applets/, 2001. Retrieved December, 2002.

[21] D. Zollman. Visual quantum mechanics. http://www.phys.ksu.edu/perg/vqm/, 2001. Retrieved December, 2002.