Building Resources for Teaching Computer Architecture Through Electronic Peer Review

Edward F. Gehringer
Depts. of ECE and Computer Science
North Carolina State University
efg@ncsu.edu

1. Abstract

Electronic peer review is a concept that allows students to get much more feedback on their work than they normally do in a classroom setting. Students submit assignments to the system, which presents them to other students for review. Reviewer and author then communicate over a shared Web page, and the author has a chance to submit revised versions in response to reviewer comments. At the end of the period, the reviewer gives the author a grade. Each author gets reviews from several reviewers, whose grades are averaged. At the end of the review period, there is a final round when students grade each other's *reviews*. Their grade is determined by the quality of both their submitted work and their reviewing.

This paper reports on our use of peer review in two computer architecture courses, a microarchitecture course and a parallel-architecture course. Students in these courses engaged in a variety of peer-reviewed tasks: Writing survey papers on an aspect of computer architecture, making up homework problems over the material covered in class, creating machine-scorable questions on topics covered during the semester, animating and improving graphics in the lecture presentations, and annotating the lecture notes by inserting hyperlinks to other Web documents. Students generally found these exercises beneficial to their learning experience, and they have provided resources that can be used to improve the course. In fact, with such a system, large classes are actually a blessing, since they produce better and more copious educational materials to be used in subsequent semesters.

2. Peer Review in the Classroom

Peer review is a concept that has served the academic community well for several generations. Thus, it is not surprising that it has found its way into the classroom. Dozens of studies report on different aspects of peer review, peer assessment, and peer grading in an academic setting. A comprehensive survey can be found in Topp 98. Experiments with peer assessment of writing go back more than 25 years [4]. Peer review has been used in a wide variety of disciplines, among them accounting [8], engineering [7, 10], mathematics [3], and mathematics education [6].

However, *electronic* peer review experiments have been much rarer. Although the Daedalus Integrated Writing Environment [1] is widely used for peer assessment of student writing, only a few computer-mediated peer-review experiments have taken place in other fields. An early project in computer-science and nursing education was MUCH (Many Using and Creating Hypermedia) [9, 11]. The earliest reported software program to support peer evaluation was evidently created at the University of Portsmouth [12]. The software provided organizational and record-keeping functions, randomly allocating students to peer assessors, allowing peer assessors and instructors to enter grades, integrating peer- and staff-assessed grades, and generating feedback for students. One of the early Web-based peer-review experiments was described by Downing and Brown [2]. Their psychology students collaborated to create hypertexts which were published in draft on the World Wide Web and peer reviewed via e-mail. Our project was one of the first to use the Web for both submission and review of student work.

3. Peer Review on the Web

There is much to recommend a Web-based approach to peer review. Unlike software that is written for a specific academic field (e.g., English composition), a Web-based application can accept submissions in practically any format, including diagrams, still pictures, interactive demonstrations, music, or video clips. Of course, the student has to understand how to produce such a submission, but for each field, that expertise tends to "come with the territory."

Secondly, the Web is a familiar interface. Most students use the Web in their day-to-day studies, so they can pick up a Web-based application for peer review with minimal effort. In addition, many if not most students are already familiar with tools for producing Web pages; for example, almost all wordprocessors can save files in HTML format.

Thirdly, Web creation skills are of increasing importance in business as well as academia. In producing work for Web-based peer review, students not only learn about the subject of their submission, but also gain valuable experience with software they will use in their later studies and on the job.

Fourthly, a Web interface enables the peer-review program to be used in distance education, which is an important and rapidly growing segment of the education market. On-campus students can review distance-education students, and vice versa, bringing the two groups closer together in their educational experience. With Web-based submission, there is no extra overhead for the instructor or TAs in handling distance-education students.

Finally, Web-based peer review facilitates the production of Web-based resources. The best peer-reviewed work can be turned into materials to help future classes learn. For example, students can create machine-scorable questions for each lecture, with different sets of students choosing different lectures. The best questions on each lecture can be incorporated into daily quizzes delivered via a Web-based testing system such as LON-CAPA [15], Mallard [16], or WebAssign [17].

Or, students can write research papers on various topics assigned by the instructor (e.g., the branch-prediction strategy of a particular processor architecture). The best paper on each topic can then be presented to the next semester's students as background reading on that topic. The writers can be asked to include liberal doses of hyperlinks in their papers, so that later students can read not only their work, but also the analyses of experts.

Figure 1. PG's welcome page

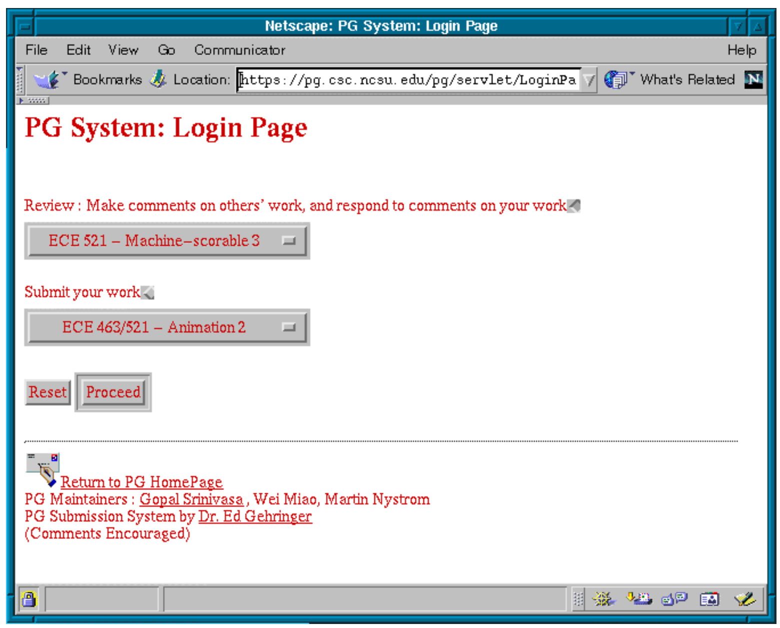

Figure 2. PG's login page

4. The PG System

PG (Figure 1) is a Web-based application for peer review and grading. It is written in Java and is servlet based. Students submit their work over the Web. Reviewers can be assigned pseudo-randomly by PG, or by the instructor, using a spreadsheet. The number of reviewers is arbitrary, but usually three or four students are assigned to review each submission. Reviewers and authors communicate double-blindly via a shared Web page. At the end of the review process, the reviewer assigns a grade to each author whose work (s)he has reviewed. A student's grade is the average of the grades given by the reviewers, plus an incentive described below to encourage careful reviews.

A student entering the PG system (Figure 2) has a choice of whether to submit a new page or review pages submitted by others. If more than one Web page is to be submitted, they may be submitted sequentially, each with a different filename, or submitted in a single Zip or tar file, which PG will unpack into its components. Entire directory hierarchies may be submitted in this manner. Since the files themselves are copied, all work to be reviewed will have a URL beginning with the pathname of the PG system, not the submitter. This ensures that the reviewers will not be able to guess their authors' identities by dissecting the URL. The ability to submit directory hierarchies allows large projects to be submitted as easily as small ones.

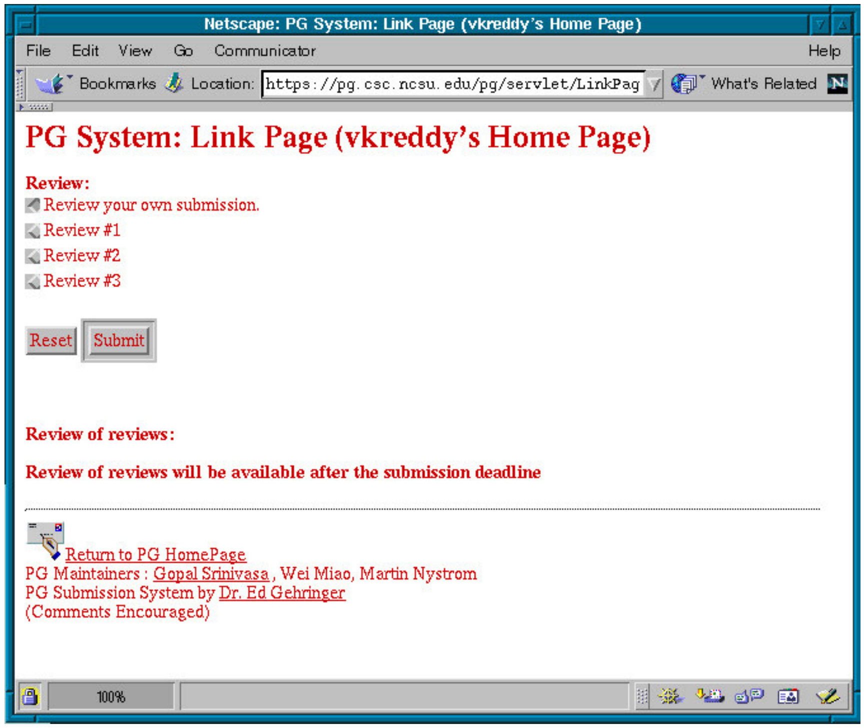

Figure 3. Page with links to submissions to be reviewed

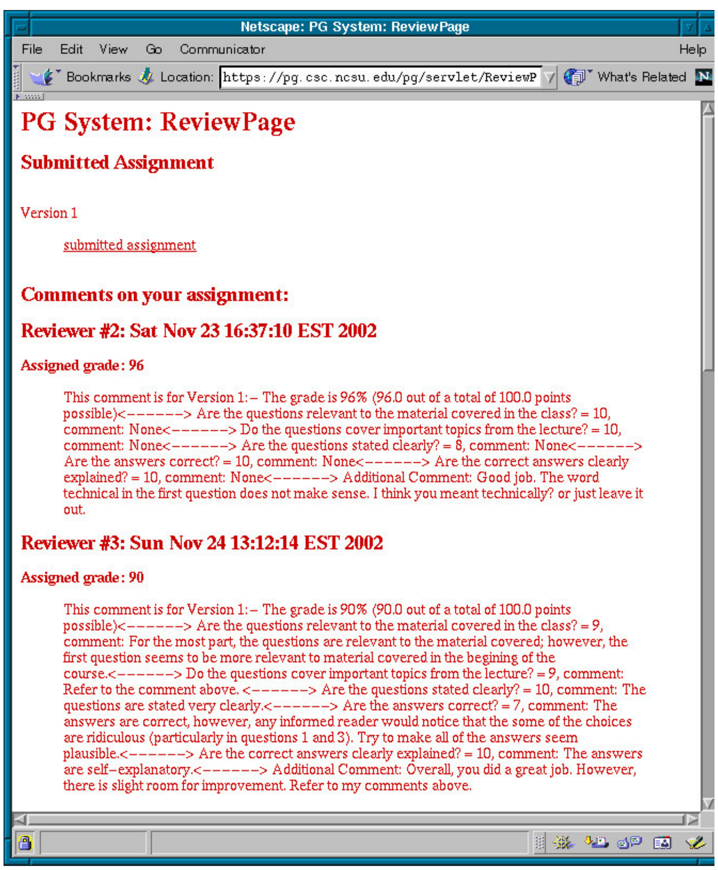

Figure 4. Review page

Reviewers communicate with their authors via a shared Web page. There is one such page for each author (Figure 3); the author can view the reviewers' comments and vice versa. The instructor can configure the system either to allow (Figure 4) or not to allow reviewers to see the other reviewers' comments and assigned grades. There are reasons in support of both strategies. Allowing reviewers to see each other's feedback provokes better dialogue over the quality of a submission, but the first reviewer's comments may unfairly influence subsequent reviewers' assessments.

Grading is based on a *rubric* consisting of several questions that the reviewer must answer with a numeric score. The questions may be assigned different weights, if desired. The grade that a particular reviewer gives a student is calculated by summing the product of each score with the corresponding question weight. A rubric-oriented approach is used to insure that all students are graded on the same criteria, and to reduce the chance that a reviewer will give an unrealistically high grade due to ignoring some of the criteria that the submission is supposed to meet. In addition to giving numeric scores, the reviewer has ample opportunity to give feedback to the student on how to improve. This can be seen in Figure 5.

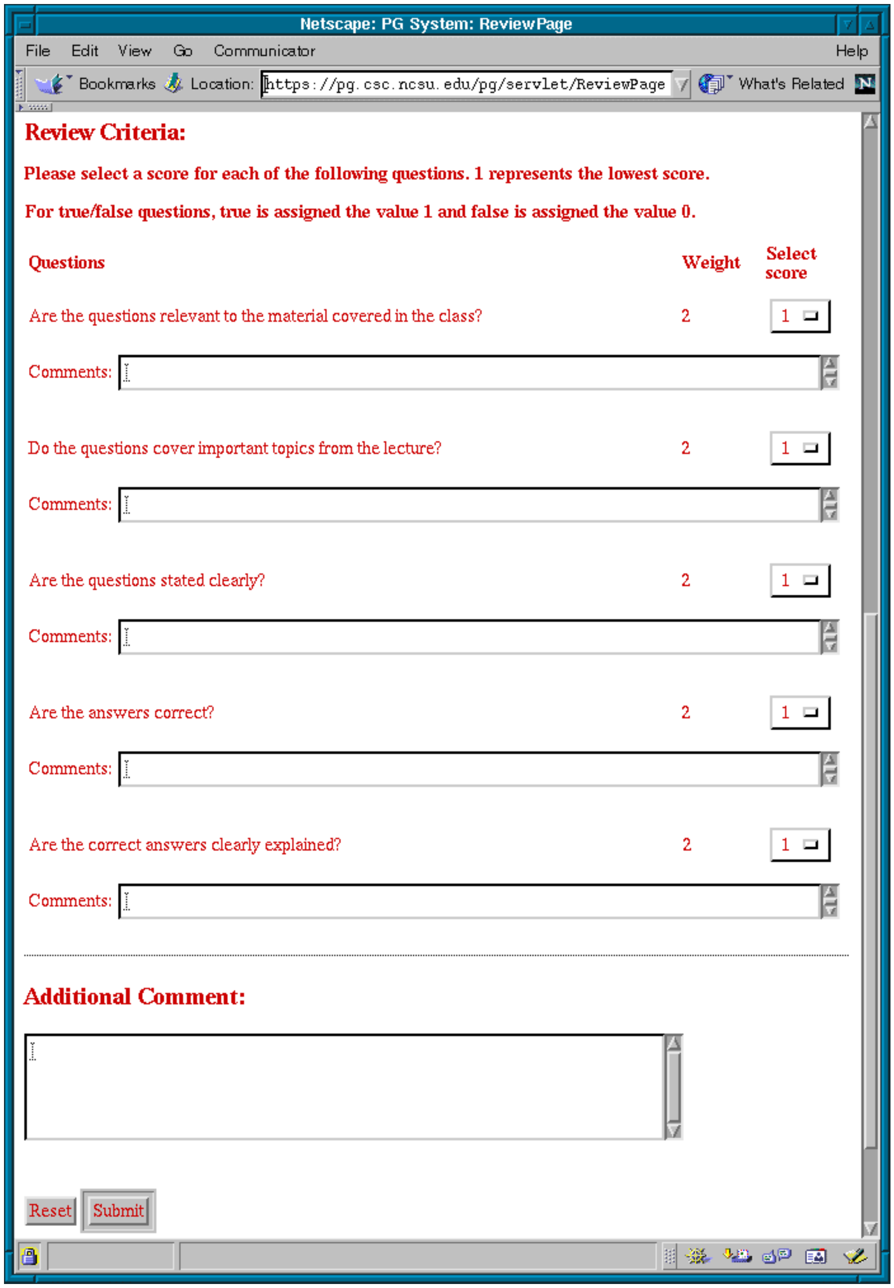

Figure 5. Grading rubric

5. The Submit-Review-Publish Cycle

Our experience with PG has led us to a four- to six-phase cycle, capable of producing high-quality peer-reviewed work suitable for Web publication.

1. The *signup* phase (optional): If not all students are to do the same assignment, the students are given a list of potential topics (relating to research, or to a particular lecture, etc.) and sign up for one of them. To assure that all topics are chosen, only a limited number of students is allowed to sign up for any particular topic.
2. The *submit* phase. Students prepare their work and submit it to PG.
3. The *initial feedback* phase. Students are given a certain period of time—usually 3 to 7 days—to make initial comments on all the work This phase was instituted after students complained that their reviewers often did not comment on their work until it was too late to revise it. Reviewers may assign a grade during this period, but they are not required to do so.
4. The *grading* phase. During the next period—again usually 3 to 7 days—students can revise their work in response to reviewers' comments, and reviewers can comment on the revisions. At the end of this give-and-take, reviewers are required to assign a grade. This grade is one component of the author's final grade for the assignment.
5. The *review of review* phase. After the review period is over, each student is presented with a set of reviews to assess. The students grade each *review* based on whether it was a careful and helpful review of the submission. The grades the students receive on their *reviewing* is then factored into their grade for the assignment (usually 25% of their grade is based on their reviewing). This phase was instituted

after it was discovered that many students were doing cursory reviews. As will be seen in Section 7, this is a sufficient incentive to be careful in reviewing.

6. The *Web publishing* phase (optional). PG creates a Web page with links to the best student assignment in each category. As described below, this can serve as a useful study tool for future generations of students.

6. How Peer Review Has Been Used in Computer-Architecture Classes

There are opportunities to use peer review in almost any course. One of the best opportunities is in evaluating student writing. Prospective employers and thesis advisors widely believe that technical students need frequent opportunities to hone their writing skills. But students need ample feedback in order to improve. Peer review can give more copious feedback than instructor or teaching-assistant review, for the simple reason that each student has only a few submissions to review, rather than several dozen. Moreover, students will be writing for an audience of their peers later in their careers, so it is important for them to learn how to do this.

In computer-architecture courses, I have assigned students to write **reviews of papers** from the technical literature. I always assign two or three related papers so that the students cannot simply summarize a paper, but must instead integrate material learned from different sources. For example, in my microarchitecture course (using the Hennessy-Patterson text *Computer Architecture: A Quantitative Approach*), I assigned these papers on power-aware architectures:

> "Energy-effective issue logic," Daniele Folegnani and Antonio Gonzalez, *28th International Symposium on Computer Architecture*, July 2001, pp. 230-239.
>
> "Drowsy caches: simple techniques for reducing leakage power," Krisztian Flautner, Nam Sung Kim, Steve Martin, David Blaauw, and Trevor Mudge *29th International Symposium on Computer Architecture*, May 2002, pp. 148-157.

I also have students do **annotations** of my lecture notes, which are on line as PowerPoint or Word files [13]. Each student signs up to annotate one of the lectures during the semester. Depending on how many students there are in the course, two to four annotations of each lecture are produced. This consists of inserting hyperlinks to other Web pages that define the term or describe the topic I am covering. Typically the students insert several dozen hyperlinks in each 75-minute lecture. The best annotation of each lecture (the one with the highest grade) is then made available to students in the next semester. In this way, students in one semester produce a resource that helps students in subsequent semesters to fill in gaps in their understanding of the material.

An excellent way to improve students' understanding of the material is to have them make up questions over what they have studied. I have assigned two different kinds of peer-reviewed questions. The first is **madeup homework problems**. Students are asked to make up a problem similar to those on the problem sets I assign for homework (typically these are problems from the textbook or similar problems). The students then peer-review each other's problems. Students learn by checking each other's work, and the problems they make up are often good enough to be used for subsequent homeworks and exams in the course. For example, in the last three times I've taught my parallel architecture course, I've used 27 problems that were made up by students in previous semesters. Given the fact that most instructors say [14] it is either important or very important to increase their supply of questions beyond what they now have, the usefulness of this approach cannot be denied.

Students' comprehension of lectures can be improved if they are asked a set of questions about the lecture after viewing it. In recent years it has become possible to pose questions and score student answers via a Web assessment and testing system like LON-CAPA [15], Mallard [16], or WebAssign [17]. It would be a major time commitment for the instructor personally to write a set of questions on each lecture, but peer review makes it possible for the students to write the questions themselves. Moreover, these questions come already "pretested" by a small set of students—the peer evaluators. Beginning in Fall 2002, I had students write a set of **machine-scorable questions** over the material in a specific lecture. This produced a "bank" of questions that can be used to create daily quizzes for students in later semesters. Ultimately, they could become a resource for a Web-enhanced version of the textbook we are using.

Computer architecture is a rather visual subject—one's comprehension is often improved by seeing a picture, or a graphical simulation, of a topic or an algorithm. Cache coherence and instruction-level parallelism are examples of such topics. Since some students are gifted in visual arts, I have allowed students to choose an **animation** as one of their peer-reviewed assignments. The best of their animations can then be incorporated into future lectures.

Peer review can be used for **research papers**. Though I have not yet assigned this in a computer-architecture course, in my operating-systems course, I had each student select a research topic from a set that included topics like “Scheduling in Windows NT,” “Deadlock handling in Unix or a particular flavor of Unix,” and “Virtual memory in Linux.” Similar topics in architecture would be the cache-coherence algorithm, branch predictor, or instruction-retirement approach used by a particular architecture.

Assuming that students have the requisite computer skills, electronic peer review is as widely applicable as peer review in general. The author has previously reported on its use in computer science [18] and ethics in computing [19] courses.

Through peer review, each class can stand on the shoulders of previous classes, learning the material with better resources, and producing ever-better tools to teach future classes. In some cases, instead of seeing large classes as a burden, an instructor may come to prefer them because they can create more formidable Web-based resources, and do so without burdening the instructor and with additional grading responsibility. This is an example of “education engineering” [20]—developing methodologies and tools to create educational materials more quickly and in greater volume, and disseminate them without loss of quality to the increasing numbers of students seeking a technologically up-to-date education.

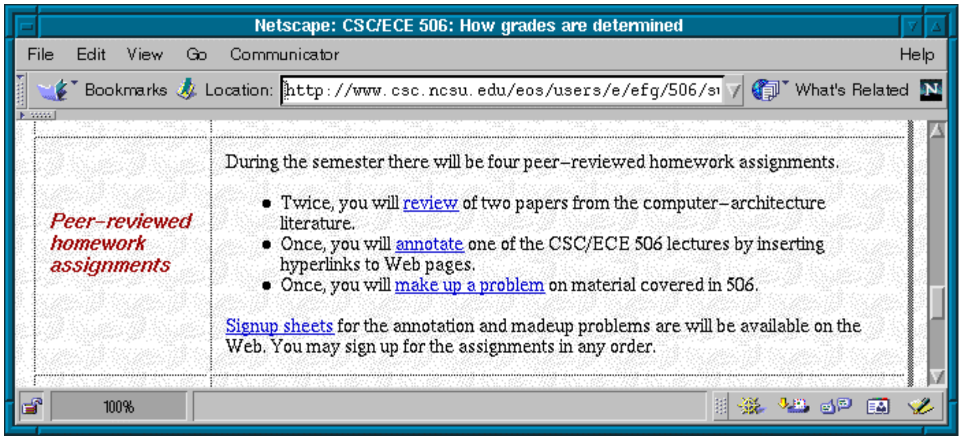

Figure 6. Peer-reviewed assignments in parallel-architecture class

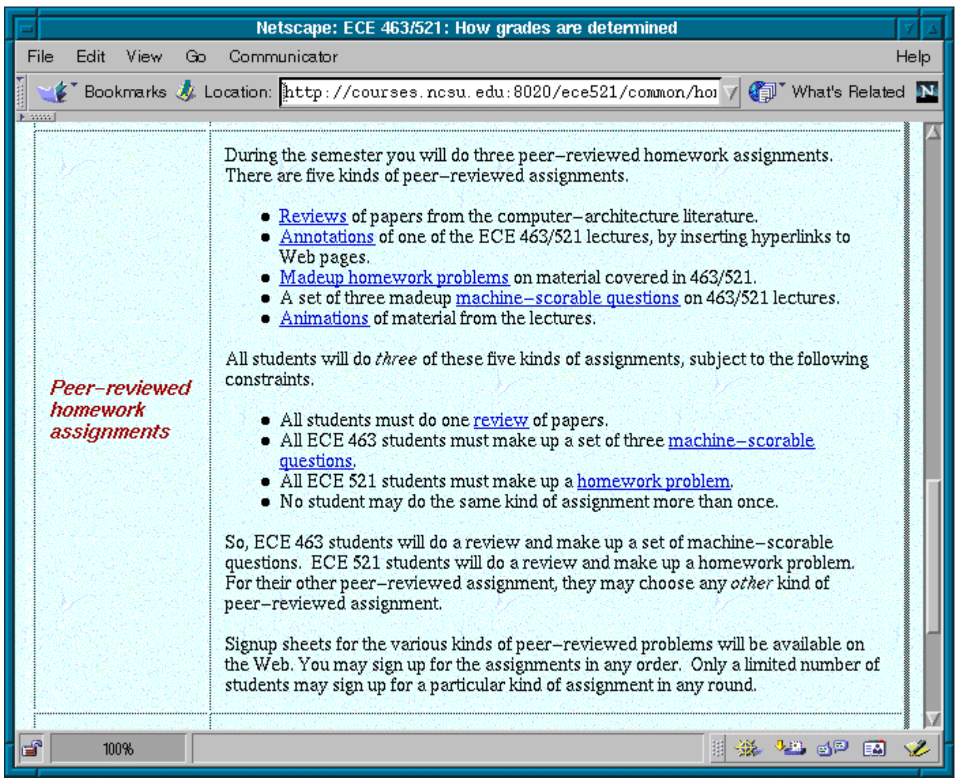

Figure 7. Peer-reviewed assignments in microarchitecture class

7. Choosing Assignment Types

During the semester, I assign several peer-reviewed assignments, and several types of peer-reviewed work. I give the students a choice of which order to do the assignments, subject to the constraint that there is a limit to the number of students doing each type of assignment for each deadline. This strategy is motivated by a desire to students doing all kinds of assignment soon after each lecture, so that, e.g., while Lecture 10 is fresh in their minds, some students will be making up problems, some will be annotating, and some will be creating animations. This insures that I get problems, animations, etc. over a wide range of lectures, rather than having all submissions concentrated on the lectures that were covered near the time an assignment was announced.

Figures 6 and 7 show the assignments I recently gave in my masters-level parallel-architecture class (CSC/ECE 506) and my combined senior/masters-level microarchitecture class (ECE 463/521).

8. Student reaction

Students in both architecture classes were surveyed at the beginning of January 2003. In CSC/ECE 506, 16 of 36 students responded, a rate of 44.4%. In ECE 463/521, 71 of 96 students responded, a rate of 74.0%. The classes did not vary much in their reaction.

The comments provided by students indicate fairly strong support for the concept of peer review, but they take issue with three aspects of the way it was implemented for these two courses.

- They thought they were hurt by the fact thbat a few students did not do their reviews. In fact, the version of PG used at the time did not deduct points for students who failed to do their reviews. During Fall 2002, PG was modified to do this checking, and it will be in the system in coming semesters. This should lead to more reliable reviewing and therefore address this criticism.

- While generally supporting the idea of multiple review deadlines, they sometimes submitted an update that was never re-reviewed by their reviewers. Currently, there is no guarantee that a reviewer won't complete reviewing in Round 2 before an author resubmits. The scheme will be changed in Spring 2003 to have extra deadline, so that there is a review period followed by a resubmission period, followed by a second review period. This should take care of the problem.

- A number of students objected to reviewers who gave low grades but few if any suggestions on how to improve. Now students are told, during the review-of-review period, to downgrade reviewers who deduct points without explaining why.

Note that, in general, students thought that reviews of reviews *were* effective (3.9 on a scale of 5) in motivating careful reviews. This suggests that giving students some guidance in how to evaluate reviews can motivate students to review according to guidelines that they are given.

9. Conclusion

Electronic peer review has proved to be an effective technique for teaching computer architecture. It allows the students to get experience writing for their peers, and it facilitates the production of educational resources that can be used by future classes, such as annotated lecture notes, homework and test questions, and daily machine-scorable quizzes. However, effective implementation of peer review is tricky. Reviewers must be given good guidance in how to review and sufficient motivation to do a good job. Authors must be given enough time to revise their work pursuant to reviews, and reviewers must be given enough time to complete their final pass. Our experience with PG has given us many ideas on how to improve the process and outcomes of peer review.

Table 1. Student Evaluation of PG

		CSC/ECE 506	ECE 463/521
1	Peer review is helpful to the learning process.	3.63	3.41
2	I was satisfied with the reviews of my work.	3.53	3.47
3	The feedback I obtained from the reviews helped me to improve my work.	3.60	3.49
4	Two review deadlines were imposed, one for the first review and another for the final grade. Did this provide an adequate opportunity for you as an author to respond to the comments of your reviewers?	3.60	3.83
5	The knowledge that my reviews would be reviewed motivated me to do a careful job of reviewing.	3.93	3.92

10. References

[1] The Deadalus Group, Daedalus Integrated Writing Environment, http://www.daedalus.com/info/overtext.html

[2] Downing, T. and Brown, I., "Learning by cooperative publishing on the World-Wide Web," Active Learning 7, 1997, pp. 14-16.

[3] Earl, S. E., "Staff and peer assessment: Measuring an individual's contribution to group performance," *Assessment and Evaluation in Higher Education* 11, 1986, pp. 60-69.

[4] Ford, B. W., *The effects of peer editing/grading on the grammar-usage and theme-composition ability of college freshmen*. Dissertation Abstracts International, 33, 6687.

[5] Gehringer, Edward F., "Strategies and mechanisms for electronic peer review," *Proc. Frontiers in Education 2000*, Kansas City, October 18–21, 2000 (to appear).

[6] Lopez-Real, Francis and Chan, Yin-Ping Rita, "Peer assessment of a group project in a primary mathematics education course," *Assessment and Evaluation in Higher Education* 24:1, March 1999, pp. 67-79.

[7] MacAlpine, J. M. K., "Improving and encouraging peer assessment of student presentations, *Assessment and Evaluation in Higher Education* 24:1, March 1999, pp. 15-25.

[8] Persons, Obeua S., "Factors influencing students' peer evaluations in cooperative learning," *Journal of Business for Education*, Mar.–Apr. 1998.

[9] Rada, R., Acquah, S., Baker, B., and Ramsey, P., "Collaborative learning and the MUCH System," *Computers and Education* 20, 1993, pp. 225-233.

[10] Rafiq, Y., & Fullerton, H., "Peer assessment of group projects in civil engineering," *Assessment and Evaluation in Higher Education* 21, 1996, pp. 69-81.

[11] Rushton, C., Ramsey, P., and Rada, R., "Peer assessment in a collaborative hypermedia environment: A case-study," *Journal of Computer-Based Instruction* 20, 1993, pp. 75-80.

[12] University of Portsmouth, "Transferable peer assessment," in National Council for Educational Technology [ed.], Using information technology for assessment, recording and reporting: Case study

[13] http://courses.ncsu.edu/ece521/common/lectures/notes.html

[14] Gehringer, E., "Reuse of homework and test questions: When, why, and how to maintain security?" submitted to ASEE 2003 Annual Conference, Educ. Res. & Methods Division.

[15] The Learning Online Network with CAPA, http://www.lon-capa.org.

[16] Mallard: Asynchronous Learning on the Web, http://www.ews.uiuc.edu/Mallard/.

[17] WebAssign, http://webassign.net.

[18] Gehringer, E., "Peer review and peer grading in computer-science courses," SIGCSE 2001: Thirty-Second Technical Symposium on Computer Science Education," Charlotte, Feb. 21–25, 2001, pp. 139–143.

[19] Gehringer, E., "Building an Ethics in Computing Website using peer review," 2001 ASEE Annual Conference and Exposition, Session 1461.

[20] Gehringer, E., "A Web-Based Computer Architecture Course Database," 1999 ASEE Annual Conference and Exposition, Session 3232.