A Scalable Capstone Course for Academic Preparation

William G. Griswold
Computer Science and Engineering
University of California, San Diego
wgg@cs.ucsd.edu

ABSTRACT

Computer science departments at research universities face on-going challenges in providing a top-quality education at a reasonable cost to large numbers of students. Moreover, students, both undergraduate and graduate, are in need of more preparation to possibly continue their academic careers at the next level as both researchers and teachers.

This paper considers the opportunity to adapt a capstone design course to meet these interwoven needs. The key idea is to employ a "hierarchy of learners" education approach, which puts all participants—undergraduates, graduate students, and faculty—in the dual roles of learners and teachers, thus preparing them for the next step in their careers. Such a course design is motivated and two variants of its implementation are discussed and evaluated, with recommendations for further improvements in scalable academic capstone in computer science.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education

Keywords

Capstone, Ubiquitous computing, Academic preparation

1. INTRODUCTION

During the last decade, UC San Diego's Computer Science and Engineering Department (UCSD CSE) has grown considerably, and we now serve over 1000 majors with 45 faculty. One of the stresses that comes with such growth is maintaining a rigorous curriculum and an individualized student experience that is up to the needs of the next generation of technology leaders. CSE met California's rapidly growing demand for degrees in computer science by increasing class sizes and allocating more graduate and undergraduate teaching assistants to help professors teach those classes. This approach has been successful to a degree, but the students did not get a personalized experience with extensive faculty contact. Nor did it permit the teaching of an adequate number of 'capstone' design courses, which have become a model for inspirational engineering education that incorporates many of the ineffables of engineering, such as teamwork. The small size of these courses have limited their availability to a select few. Lastly, it left a chasm between our undergraduate and graduate programs, the former focused largely on learning computer science subject matter, the latter on computer science research. This experience gap left our undergraduates largely unaware of the opportunities and rewards in graduate education, and somewhat unprepared to accept the unique challenges of research. Similarly, our graduate students were getting limited experience in preparation as teachers and mentors, mostly lecturing in discussion sections and working with students one-on-one in the lab.

In the following, I introduce a capstone course for academic preparation, and then a curricular innovation that I call *hierarchy of learners* that can open a capstone design course to many more undergraduates, while also preparing them for graduate school and at the same time improving the preparation of our graduate students as well. Quantitative and qualitative measures show considerable success and provide unique insights. A number of future improvements are also identified.

2. BACKGROUND

UCSD CSE's challenge is not unique. The 1998 Boyer Commission on Educating Undergraduates in the Research University issued a report detailing the challenges of large-scale high-quality undergraduate education, and proposed a solution based on existing curricular models [2]. The Boyer Commission's recommendation can be summarized as a *hierarchy of learners* approach. The idea is simple: in order to sustain itself, a research university must train the next generation of educator–researchers. The education that undergraduates and graduate students receive should prepare them to make the choice to continue their education, and give them the experience and intellectual tools to succeed if they should so choose. (Of course, such tools are also valuable outside the university. For example, undergraduates should be prepared to go on to graduate school, not just with knowledge of the discipline, but also background in teaching and research that prepares them for graduate work as well as the choice to pursue it.

The Boyer Commission's recommendation, then, is to give students first-hand experience in performing at the next level. They recommend a scalable hierarchical approach, with faculty preparing graduate students to be faculty, and graduate students preparing undergraduates to be graduate students. Moreover, they recommend that more senior undergraduates prepare more junior undergraduates. A concrete instantiation of these recommendations is for a course to employ a professor, a few graduate students, and several more senior undergraduates in teaching. Typically, the professor will teach the main lecture, graduate students will conduct

ACM SIGCSE 2005

discussion sections, and the undergraduate assistants will help students in the lab. To maximize faculty and graduate student contact with the students, they not only hold office hours, but also spend time in the lab with the students.

CSE in fact already implements this approach for most of its core courses, but both peoplepower (graduates and undergraduates) and the funds to employ them are limited. This is adequate for many courses, but not for capstone design courses, in which undergraduates demand and deserve continuous intimate interaction with the professor and assistants. The Boyer Commission counsels that the way to address this is to treat the graduate student and undergraduate assistants as students in the course, *in the role of learning to teach computer science*. Of course, "the best way to learn a subject is to teach it," so these students are learning in multiple dimensions.

3. CAPSTONE FOR ACADEMIC PREP

My first step in addressing this need through the Boyer Commission's insights was to develop a capstone course that met our students' needs for academic preparation. For this purpose, I created Applications in Ubiquitous Computing, designed and taught in Fall 2002 (http://www.cse.ucsd.edu/users/wgg/CSE118/index-2002.html). The class had two elements, a research project and twice-a-week 80 minute meetings in the form of a roundtable discussion of one or two research paper. The discussion was facilitated by myself. Although the research project contains the essential seed of a capstone design project, there are considerable differences in "capstoning" for academic preparation. The research project is designed as an opportunity to conduct research like a graduate students do, albeit on a tighter schedule. To formulate a research proposal, the students need background in not only the subject matter of ubiquitous computing, but also the nature of the research in the field. Hence, the use of research papers for conveying subject matter, rather than a text. To further sensitize the students to the research context, several of the roundtable discussions were replaced by colloquium talks from outside researchers, guest speakers (both faculty and graduate students), and a visit to a research lab on campus (not mine). One class meeting was dedicated to how to prepare a research project proposal, and another for presenting and critiquing the teams' proposals. At a separate "conference" held at the end of the quarter, each team publicly presented its project in the form of a research talk. If the project involved building or extending a system, the presentation included a demonstration of the system.

The students were intentionally challenged in two ways in this course. First, they were unfamiliar with critically reading and discussing research papers. As part of the introductory class, I gave substantive guidance, captured in this web document: http://www.cse.ucsd.edu/users/wgg/CSE210/howtoread.html. The core of this document is a form that the students are expected to fill out for each class that helps them answer the four key questions for almost any engineering paper: What are the problems this paper means to address? What is the proposed solution? How is it evaluated and what is the result? And what are the implications, including future work? The rest of paper reading was learned "on the job", so to speak, which is the learning modality counseled by the Boyer Commission and consistent with capstone design. Second, students were of course unfamiliar with formulating and proposing a research project, hence the special class session on that topic, the public research proposals, and considerable interaction between me and the project teams. As an additional aid, the students were encouraged to write proposals that answered the questions on the form they filled out for each class period. This also reinforced the message that they were working in the same sphere as the researchers they were studying.

The choice of embedding this material in a class on ubiquitous computing, and as a capstone design course at that, was carefully considered. Ubiquitous computing is a relatively new and "hot" subject in computer science [6], and also rather interdisciplinary (e.g., systems, software engineering, HCI, even social implications), and hence is attractive to many students. It is also rather accessible, being oriented towards applications. Such attractive features are especially necessary in overcoming some of the preconceptions about academic preparation. It also has the advantage that there is no definitive text yet, so engaging the research literature is a natural adjustment. Also, ubiquitous computing as an applied topic lends itself well to capstone design, and the "learning by doing" element of capstone design lends itself well to academic preparation. Certainly, sitting through lecture after lecture on "how to do research" would send the wrong message about the delights of graduate school. Learning how to do research as a side-effect of doing research in ubiquitous computing is natural, realistic, and fun.

Even at a small scale—16 students were selectively enrolled—the course has elements of the hierarchy of learners approach, as the students, in part through peer learning, got an intimate look at—and preparation for—graduate student life. Of the eight students who were graduating that year, I wrote graduate school recommendation letters for four of them. These letters were substantive and strong due to the rigor of the course and the intimate seminar format. The course and professor received 100% ratings in the standard student-administered course review.

Yet, the course could not succeed unchanged at a larger scale. Indeed, having 17 people (including myself) in a roundtable discussion was imperfect. Although this works at the graduate student level, the undergraduates seemed to need a smaller group. If everyone participated equally, there were not many opportunities each class period to practice speaking about the research topic. And often, half of the group spoke much more often, further depriving the other half.[1] Also, I found it difficult to adequately mentor many project groups, and discouraged smaller groups from forming even if they made sense for the students and the project. And, time considerations meant that the "academic preparation" element of the course was largely implicit—there were no in-class discussions about career choices, how research universities work, etc. Lastly, the students had only two (unsustained) exposures to graduate students—their possible future colleagues—during the course: as a visiting lecturer and during the visit to the research lab.

4. SCALABLE CAPSTONE

The question, then, is how such a course's impact could be increased both by including more undergraduates of wider ability—CSE was hoping for 60 students—as well as some graduate students. By including graduate students, the quality of the course be higher both because of increased attention on the undergraduates and exposure to their cohort. The graduate students could benefit, too, if the course provided them experience in teaching and mentoring graduate students. The following presents how I taught the second instantiation of this course under these assumptions. For a variety of reasons—such as an "early morning" (9:30am) meeting time—the enrollment of the course was just 17 students (and 5 graduate students enrolled in a parallel graduate course). Yet, as already argued above, even at this size, with relaxed admission criteria, substantive changes were required (http://www.cse.ucsd.edu/users/wgg/CSE118/index-2003.html).

Lecture Series. First, with relaxed selectivity, some students would

[1] I recognize that the quieter students may be learning much by listening. It's fine that they listen, too, in smaller groups, but less than ideal if they choose to listen because the group is relatively large.

need—and all would benefit from—specific training in software engineering and project management. Consequently, I introduced a once-a-week lecture, given by myself, teaching the basics of software engineering, teamwork, and project management. Software engineering and project management are best conducted in a domain-specific manner [3], so motivating background, methods, so examples could be developed in the Ubiquitous Computing context. Due to the inherent uncertainties faced in research, I taught software engineering through a variant of Extreme Programming [1, 3]. Such a lecture series is common in capstone design (e.g., http://www.jacobsschool.ucsd.edu/TIES/). However, because this course was for academic preparation, there were several unique elements. First, I gave a lecture on the role of research universities in society, based on the wonderful book *The University – An Owner's Manual* [5], while also addressing issues of graduate student life. I also moved the lecture on the conduct of research and how to write a research proposal from the twice-a-week meetings to this lecture slot, and added a presentation on research presentations towards the end of the quarter. It was important that the students had previously seen an actual research presentation in the twice-a-week track.

Graduate-Directed Project Mentoring. Second, I assigned two graduate student "mentors" to each project team. These mentors were explicitly not managers. They observed and provided advice. They also served as "friendly spies", providing information to me so that I could advise the mentors and provide my own guidance to project teams as deemed necessary. This is the first element of the hierarchy of learners approach. By delegating mentorship to the graduate students: (1) the undergraduates got more attention, (2) the mentorship was high quality because the professor is overseeing it, and (3) the graduate students were getting valuable experience and guidance in mentorship of research.

Graduate-Facilitated Breakout Discussion Groups. Third, holding a roundtable discussion with up to 60 students in the room is impossible. Building on the Boyer Commission's ideas, I broke the class into two stable groups of 8 and 9 students, each facilitated by a graduate student. The graduate students, having taken research seminars before and working with a small group of students, were well equipped to succeed. With more than N/8 graduate students enrolled, a rotation was set up to allow the graduate students to observe each other.

Graduate Student Mentoring. Finally, to help the graduate students in their two new roles, I met with them a couple of days before each discussion, as well as discuss the research projects. This meeting had several elements. First, we would conduct a post mortem of the previous discussion as a way of extracting lessons learned and advancing facilitation skills. It also served to "debug" the discussion groups, as they had their own unique personalities and facilitation needs. Second, we prepared for the upcoming discussion. Generally, this took the form of myself conducting a facilitated discussion of the paper in question. From that we extracted key issues, questions for discussion, and possible pitfalls. True to the facilitation format, I did not deliver these to the graduate students, but facilitated their emergence. Since I have found it difficult myself to facilitate discussions with just a list of topical questions, I enforce a meta-structure with the graduate students that covers, in order, the four questions on the form discussed in Section 3: problem, solution, evaluation, implications. This avoids the pitfall of jumping to "dessert" without substance, in particular personal evaluation and implications of the work. Typically, I did not identify the facilitators for a given day until after this discussion, as I wanted everyone working equally hard on preparation. Finally, we would end the meeting by discussing on-going project issues. Earlier in the quarter, before the projects begin, this "project time" is spent on pedagogy, both philosophy and implementation. There are also a few twice-a-week meetings that have no facilitation, and these meetings are used for similar purposes. At the beginning of the term, before the graduate students had seen my pedagogical techniques, I performed a facilitation with the undergraduates as one large group, with the graduate students observing. (With 60 undergraduates enrolled, I would likely choose a subset of the class for this exercise, with the rest observing.)

In my graduate seminars, I keep the discussion format fresh by varying it in entertaining ways. In the undergraduate course, I have varied it in ways that highlight other aspects of the research experience. For one, I try to get three guest research speakers from three different careers: a graduate student, a faculty member, and someone from industry. I strive for each to also broaden the subject matter of the course or style of research, for example getting a speaker from the systems field, since the papers are heavy on HCI and applications. Second, we visit one of the larger and more diverse research labs on campus (http://cvrr.ucsd.edu), which includes smart rooms, smart cars, and robots. Third, I give a classic 25 minute conference research talk, with accompanying paper. Because the presentation is only 25 minutes, there is ample time for discussion not only about the topic but about research conferences themselves. To give a flavor for the creative aspects of research, as well as experimentation, my TA and I designed one session where the students created "PDA holder" devices that would make PDA's easier to use on cramped desks [4]. And as mentioned earlier, the one class session is dedicated to the students' research proposals.

Marking of the discussion is not the only role I played in the classroom during the discussions. For one, I also recorded the main ideas that emerged from the discussions, with some help from the TA. I would use these to produce an end-of-class summary, thus exposing each group to the other group's often-unique ideas.

Another role I played was to observe how the discussion was ebbing and flowing, and how the facilitation was contributing. A facilitator could of course be too hands on (in my opinion) at one time or another, too hands off, etc. As the term progressed, I developed a meta-facilitation style of sitting behind and to the side a facilitator for awhile. If the discussion seemed to be going awry, as it sometimes did—digressing too far and too long, for example—I could whisper a possible adjustment in the facilitator's ear or pass a note while an undergraduate was talking.

5. DISCUSSION

I did two things to assess the impact of this class organization, looking at the impacts of the organization on scalability and academic outcomes. First, I noted unique outcomes as they arose during the course. Second, I gave an exit questionnaire for both the undergraduates and graduates to complete. The class and professor received 100% ratings in the campus-administered survey. This is not too surprising in and of itself, given the small, intimate class format. Here, I focus on the qualitatively unique aspects of the course organization and their individual contributions.

Lecture Series. The lecture series was the least well-received element of the course, with about two-thirds saying it was useful, and the rest not. Many singled out the lecture format; and I think the early class time (9am) didn't help sustain their attention. Yet, I don't think this aspect of the course can be eliminated. The material is unique, yet complementary to the other elements of the course. For example, one student said "Very good to learn about all the realities of software development." Another said, "...gave lots of useful info that can be used through my whole career." Another option, moving it to a participatory format, would require greater

preparation by the students, which they could ill-afford given their hard work on the discussions and project.

Graduate-Facilitated Breakout Discussions. Cutting the size of the discussion groups in half was effective. Even very quiet students without native language skills could find a way to make a couple of substantive comments per class meeting. I believe that the graduate facilitators also played a role here, as their presence in the circle is not as daunting as mine. Yet, I think I would strive to make the groups smaller, and maybe rotate the membership a bit. For one, I think it would increase participation of quiet students even more. Two, graduate students could get more opportunities to facilitate; something explicitly requested by a couple graduate students in the exit questionnaire. Three, fixing the discussion group membership for the entire quarter made some students feel trapped. It also made it hard to balance the size of the groups when some students missed class.

A surprising outcome of the multiple student-facilitated groups was a quantitatively high level of discussion. Because of the prior discussion and preparation with the graduate students, I came to class each day with a "checklist" of thoughts and ideas that I hoped the undergraduates would raise themselves during discussion. More often than not, they did a *better* job than the graduate students. That is, they would raise all the major thoughts and ideas the graduate students raised earlier, perhaps some of my own thoughts too, and often ones that neither the graduate students or myself had thought of. I attribute this to three causes. One, a couple of the undergraduates were very bright—smarter than myself and the graduate students. Two, by having two groups running at the same time, more ideas could get raised and discussed, and the groups had their own personalities, so did not always go down the same path. Three, since the facilitator was not a professor (unlike in the graduate student meeting), I believe the setting was a bit less intimidating and less formal, permitting freer thinking and speaking. The result, especially with my discussion summary, is that everyone was able to benefit—the undergraduates, the graduates, and myself. And since I apprised them of their accomplishments, it was also quite a boost to the undergraduates.

Although my meta-facilitation appeared effective to me, the graduate students professed that it was somewhat distracting and hard to react on the spot. It was also a bit taxing for me to do this while also marking participation and tracking the emergence of ideas. If I could delegate discussion marking to graduate students, this would be less of an issue.

The discussions were not only measurably effective, but the students unanimously approved of the format. One student wrote "Yes, it is cool. Even I am not a person who can talk well. I can get what other people think and improve how I think." Others wrote, "Yes, opened my eyes up to some cool research", "Yes, Absolutely, have never learned so many ideas that really stick through discussion ==¿ interactive learning, makes you think, also good motivation to read papers", "It was something I look forward to every week", and "taught confidence in self." One thoughtful student observed, "It helps brings up dissenting opinion of the papers and work through different views. Allows for critical thinking."

When asked about the graduate students as facilitators, about three quarters approved. One positive respondent said, "It is good to have a facilitator to keep things moving in the right direction (easier if grad student, because of hierarchy issues)." Others said, "Worked great, good experience for them", "Nice! More ideas." Two undergraduates suggested that they could do it themselves by the end of the term, an interesting proposition. One dissenter said, "Sometime. I think Prof would be much better." The graduate students unanimously like facilitation, citing the unique challenges and experience.

Graduate-Mentored Projects. The mentoring of projects by the graduate students was embraced by both undergraduates and graduates alike. The graduate students were regularly attending their meetings, typically at times and places that would have been highly inconvenient or impossible for me to attend.

One undergraduate offered, "Better than great." Another said, "Big help. They know a lot and share a lot about their experiences." Yet a few were less sanguine, saying, "Somewhat helpful but often times we're on our own", and "It wasn't very helpful since they are not aware of projects." Here likely is some variation in how each graduate student interacted with their team, as well as the teams' needs and expectations. The fact that groups felt like they were "often on their own" was a rule I set out at the beginning, but perhaps was not clear to all undergraduates.

Topical Outcomes. With all the things going on in this course, the topical content of ubiquitous computing could have suffered. When asked about the effectiveness of advancing their knowledge in UbiComp, all but 17 of 19 student respondents claimed high effectiveness. One claimed good effectiveness, and one said "would like to get more depth in subject." A positive respondent reflected, "I probably will remember the contents of the papers read for a long time because I enjoyed the experience and because the learning process was very stress-free."

Academic Outcomes. Did the class have an impact on undergraduates' intentions to attend graduate school? Nine of reported yes, five reported no. All of the yes's and all but one no implied the students were graduate-school bound. One student wrote, "COMPLETELY, grad school is for me! I hope this wasn't dessert before dinner!" Others said "...take more action into attending", "Fits my learning style", and "It seems more fun than I thought." A typical "no" wrote, "Not much. I still want to go to graduate school."

When asked about how this course prepared them for graduate school, a couple honestly answered, "Not sure. Don't know what grad school is like." Perhaps the most telling answer was one from a graduate student, who said "Yes, I wish I had done this as an undergrad." At a higher level, one undergraduate offered, "Amazing, this class shaped a large part of my future vision."

The graduate students uniformly welcomed the teaching experience, citing its uniqueness and effectiveness. Typical comments were, "It introduced a new method of teaching that was more hands off for the teacher," and "What I probably learnt most was the effectiveness of a *de-formalized* atmosphere." One student elaborated, "I was at first nervous about the idea of teaching undergrads a topic which I knew nothing about. Through this course I found that there was knowledge I gain in my undergrad/grad that I could share. I really liked being a mentor for their project. My group was receptive and I could help them.... I feel that I got a double bonus from this class. First I learned a subject which I knew very little about. Secondly, I was able to help and lead other students and learning about teaching."

A telling experience in the Fall 2003 course was that three undergraduates (separately) approached me outside of class to talk about applying to graduate school. Each asked whether, with a 3.2 GPA, they could get into a good graduate school, being under the impression that they needed a much higher GPA. With some counseling, they all applied to graduate school, and with some strategizing, two got into good MS programs and one at a top Ph.D. program. This is not surprising in retrospect, given the informal setting of the class and the subject matter of the course. Yet, in over a decade of teaching large lecture classes, despite my public declarations of interest in the students' careers, interactions like these have almost always

been at my instigation, typically by singling out top students. The role reversal in CSE 118 is encouraging.

Other Issues. Of course, there were numerous difficulties with scaling the course, especially as it was taught this way for the first time.

One challenge was grading. Since the readings and discussion were a significant portion of the course, they constituted 50% of the grade. One problem is grading the quality of a student's participation. My approach is to start by grading charitably—essentially on effort—and then increasingly grade on results, awarding one point for each comment. By results, I mean that a student's contributions must be unique and substantive, and not impede other students' contributions. This takes some judgment, but simple acknowledgments, repetitions of others' statements, and dominating the conversation are easy to detect and discount in the grading. Yet, some confirmations of others' ideas are useful—perhaps by increasing the original contributor's confidence or saying the idea in a unique or clearer way. Because students have different learning styles and personalities, not all students can be expected to participate equally. To compensate for this, I do two things. First, I cap the amount of available participation points; excess points are discarded. Second, I scale the points by taking their logarithm and then grading on a linear scale. This was constructed from experience—participation can vary by a factor of four or more, even when the students know they are being scored on participation.

A related challenge is distributing the grading across the discussion groups. Although I feel that my own grading ability is high, developing it in others takes time. If using the graduate students in the course (e.g., appropriating observers), it puts the graduate students in the conflicting positions of neutral facilitator/mentor and grader. To overcome this problem in Fall 2003, I used an experienced, part-time teaching assistant for this purpose, an affordance that will not scale to 60 students. I could grade one group, while she graded the other group. The scalable solution, I believe, is to make the grading quantitative. The basic idea is to create a chart, constructed from experience, that provides a relatively unambiguous table-driven approach to grading. In this way, the graduate students are not making decisions, per se, but just following my instructions. This eliminates the potential for inconsistency=as well as the conflict between their facilitator and grader roles. Although somewhat constraining, exceptions (if few) can be noted and discussed after class, thus adding some judgment to the grading.

Another scalability problem transcends the course—writing letters of recommendation for graduate school—something this course tends to trigger. So far, I have buckled down and written the letters, but it was a challenge, even working from samples of previous letters. Growing to 60 students will increase the load beyond my capacity. The solution I believe, once again, is to reach out to the graduate students. Working from a template, a graduate student who served as an undergraduate's facilitator and project mentor could draft an initial letter, and then work with me to finalize it. Because the reputation of the letter writer is so important, I would at least need to be a co-signer of the letter. The advantage to everyone is that more time can be dedicated to the letters, and the graduate students can gain valuable insight into an important part of the educational process.[2]

Other possible problems are the disruptions of multi-group logistics. By assigning stable discussion groups, students not only developed a rapport, but they knew on arrival that they could set the room into circles of chairs and sit down. Noise issues were mitigated by pushing the groups into the far corners of the room and keeping the circles tight. The groups not only hear each other better, both also to speak more quietly, so their voices don't carry. Surprisingly, the observers, who sat outside the circles, had few complaints about crosstalk, although they learned which spots afforded the least crosstalk and sought them out.

I've found that the informality of a course encourages openness and creativity, but the informal post mortems of the graduate student group facilitations did not work as well as hoped. Even though we met later the same day, recollections were already vague and incomplete (I can personally atest), perhaps because the discussion is somewhat unstructured. The next version of the course will use a form not unlike my research paper critique form for helping the observers (and myself, and the facilitators after class), record their insights. At a minimum, the form will serve as a checklist of things to look for during the class and provide a stable written record. To minimize the potential embarrassment or conflicts that could arise from more incisive critique, I could allow potentially difficult feedback to be provided directly to me, which I could then discuss with the facilitator as I best saw fit. In Fall 2003, there was no obvious cause for this; the graduate students were remarkably supportive. Yet, I suspect that entering the written modality could change the dynamic somewhat.

6. CONCLUSION

Capstone design provides unique opportunities for addressing undergraduate needs, such as preparation for academic careers. The *hierarchy of learners* approach, which enrolls graduate students in the learning process, can take this to a new level. Both more students can be reached and the quality of education increased, while also advancing graduate student learning and their own academic preparation. Some scalability problems remain, but their solution appears possible. An open question is how far such a system can scale.

7. ACKNOWLEDGMENTS

This work supported by a gift from the Intel California Education Team, with additional support from Microsoft and Hewlett Packard.

8. REFERENCES

[1] K. Beck. *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Reading, MA, 1999.

[2] Boyer Commission on Educating Undergraduates in the Research University. Reinventing undergraduate education: A blueprint for America's research universities, 1998. http://notes.cc.sunysb.edu/Pres/boyer.nsf/.

[3] W. G. Griswold. Teaching software engineering in a compiler project course. *ACM Journal on Educational Resources in Computing (JERIC)*, 2(4), 2002.

[4] M. Ratto, R. B. Shapiro, T. M. Truong, and W. G. Griswold. The ActiveClass project: Experiments in encouraging classroom participation, December 2002. Submission to CSCL 2003, http://www.cs.ucsd.edu/users/wgg/Abstracts/cscl-aclass.pdf.

[5] H. Rosovsky. *The University: An Owner's Manual*. W. W. Norton & Company, New York, 1991.

[6] M. Weiser. The computer for the 21st century. *Scientific American*, pages 66–75, 1991.

[2] Another thing I've found that helps with letter writing—thanks to my department colleague Geoff Voelker—is to have the undergraduate reply to, in writing, a series of information requests (e.g., grade transcript) and questions (e.g., "What was your most important experience at UCSD?").