AC 2008-1483: DAMAGE CONTROL: WHAT TO DO WHEN THINGS DON'T WORK

Edward Gehringer, North Carolina State University

Damage Control: What to Do When Things Don't Work

Edward F. Gehringer
North Carolina State University
efg@ncsu.edu

Abstract

What should you do when your software breaks, when your labs don't work out, when your tests are bombed, when your homework has errors? These are questions every instructor has faced. This paper brings together a set of tips from dozens of experienced educators in dealing with these awkward situations. They tell us that it is best to be honest with students and admit your mistakes. Students will recognize that you are human, and besides, errors and changing requirements crop up on the job too, so learning how to handle them is good experience for real life. Often a mistake in an assignment or lab can be turned into a learning experience. There are many options in dealing with test questions that don't work out for some reason. The weighting of questions can be adjusted in various ways, or homework assignments can be given to give the students a second chance to learn the material.

1. Introduction

As any new engineering educator knows, there are a lot of things that can go wrong in class. A lecture may leave a class totally confused, or totally bored. Homework assignments may take the students much too long—or they may turn out to have no solution. Labs may yield experimental results far different from what the theory predicts. Tests may be too long, or they may contain errors that waste students' time. When confronted by any of these situations, what's an instructor to do?

This paper is based primarily upon responses to questions posed on two mailing lists, the Engineering Technology listserv, etd-l@listproc.tamu.edu, serving ASEE's Engineering Technology division, and the SIGCSE members list, SIGCSE-members@LISTSERV.ACM.ORG, serving the Special Interest Group on Computer Science Education of the Association for Computing Machinery. The author posted to these lists twice, in October 2007 and again in January 2008. More than 50 engineering and computer-science educators shared tips in these discussions.[1] The published literature was also consulted, via Google Scholar and ERIC, the Education Resources Information Center database. These turned out to be much less useful sources, since it proved impossible to come up with search terms that would return information on the desired topic without also returning reams of extraneous material.

In general, when things go wrong, the best policy is to admit your mistakes: "… even professors sometimes make mistakes, and this was one of them."[2] Never try to fake the students out. As

[1] Instructors were asked if they wanted to be quoted in the paper. If so, they were asked whether their name could be used. According to their wishes, some of the quotations identify the author and some do not.
[2] Doug Baldwin, SUNY Geneseo

Molisa Derk[3] put it,

> The worst thing you can do is try to "cover up" and BS your way through it. The kids will see through it. They will forgive mistakes if you admit your error and respond fairly. If a homework is undoable, replace it with another assignment. (I have done this.) If a test question is misleading, either eliminate it from the grading, or give everyone credit for a correct answer. (I have done this.) If a test is too long or too hard (I have done this too!) add a "fudge factor" to everyone's score so that the mean and/or median is more "normal". And apologize. The students will respect you."

An ounce of prevention, though, is worth a pound of cure. Try not to be too ambitious the first time you teach a course. Your job is to teach the students, not impress them with how much you know. Ask colleagues if you can reuse[4] or adapt their lecture notes, rather than making yours up from scratch. This is not plagiarism if you have their permission and make it clear that you have adapted your materials from those used earlier in the course. Your colleagues or department may even appreciate this, because it helps to insure that all students learn a common body of material, regardless of whom they take the course from.

Be realistic in the work you assign. Limit it to what you can work through yourself. Ray Schneider[5] relates,

> [T]he first time I taught compilers, I was catching up with the material[.] I put an inordinate amount of effort in and managed to snow the students, but the tests were mostly parrot back what I'd said in class. I sometimes handle these kind of problems with take-home tests keyed to the book's problems. [W]e can discuss the problems they worked on in class and … work through the material together … [T]hey don't have to know that you never saw if before you worked through it. … [Y]ou can usually pick and choose the problems to work in class so that you pick the ones you really understand.

However, eventually you will come up to speed on the material, and look for other ways to challenge your students. Then at some point, you will probably need to engage in damage control. As Glenn Blackwell[6] puts it,

> My opinion … would be that anyone in the teaching profession for any length of time who has not run into this is teaching things too easy. Either too easy in the sense the course material should be more difficult/challenging, or being too easy on his/her students.

2. General guidelines

Whatever goes wrong in a lecture, assignment, or exam, the first rule is to find a fair resolution. To students, this means that insofar as possible, it should not affect anyone's grade adversely. It's difficult to give a general rule for how to do this, because each situation is different. One instructor explained:

> I have given tests that were too long, had unsolvable problems on tests, had problems that could have been easily perceived as misleading, chosen homework problems from the text which had typos, labs that have run amuck, computer lab snafu's, etc, etc. I just tell the students I will come up with a solution that addresses their concerns (how this problem will affect their grades) and

[3] Molisa Derk, Oklahoma City University
[4] Max Hailperin, Gustavus-Adolphus College
[5] Ray Schneider, Bridgewater College
[6] Glenn Blackwell, Purdue University

> will be fair. I have thrown out tests, redone labs, I have curved a test to address a specific problem, and I have done nothing.[7]

Usually, the simplest correction will be the most successful. If a test is too long, I can add *x* points to everyone's score without getting any complaints. But if I use a more complicated formula to prevent scores of more than 100% (which no one could have gotten if the test was short enough), it seems that someone will inevitably complain that the formula hurts them.

It is possible to extricate oneself gracefully from fairly serious situations in this way. Doug Baldwin relates an embarrassing incident:

> Actively try to correct the mistake, but in a way that won't disadvantage any student relative to what you originally intended. (As an example of this principle, I once lost a briefcase full of hour exams waiting to be graded. I figured that some students would be delighted if I just forgot that this exam ever existed, while others were probably counting on it to pull up their course grades; a replacement exam would help that latter group, but some students would find it a burden that they didn't have time to prepare for. So I ended up letting each student choose whether the take the replacement exam, or count an earlier hour exam twice in their course grade. Students were justifiably unhappy that I had lost the original exams in the first place, but generally seemed to accept that the solution was fair to all.)

In another case, it was shown that missing a question did not seem to affect the results of a test.

> I gave a test to over 60 students once. I made photocopies of 40 tests at one point, then later did 20 more. [In] the second batch, the copier took in two pages at once. The result was that 20 tests were short one question. It was a math class and there was one question per page with the rest of the page to show the student's work on the problem. No one noticed during the test. I did not know until I graded them. I did statistical analysis of both groups. The average was the same, the high and low scores were almost identical. I just explained to the students what had happened. Told the class that there was no significant difference in test scores. I told anyone who felt they had an unfair advantage to come see me in my office hours. No students did.[8]

Resolutions that allow students to earn extra points seem popular. Max Hailperin reports,

> Regarding grading, I often find it useful to retroactively change disasters into extra-credit opportunities. That is, I explain that the problem won't count against anyone's grade, but that those students who managed against all odds to solve (or make substantial headway toward solving) a very difficult (or impossible) problem will be rewarded with extra-credit points.

Students should not expect assignments to be perfect. Several instructors made the point that in the real world, requirements are incompletely specified, change during the project, etc. As Kathy Roberson[9] put it,

> Up front, and several times during the semester, I explain to my students that they are preparing for a career where they will frequently be faced with things that don't go right. Clients are notorious for asking for one thing but really needing something else. In fact, over the course of the semester, I intentionally include several assignments that are too hard, or the deadline too short, or that have conflicting information, or just don't make sense. I want my students to begin to think about what they are doing ahead of time. I want them to be able to negotiate win-win solutions that meet the needs of their clients and can be delivered with the resources available.

---

[7] This author has requested anonymity.
[8] Erin Vitale, Alfred State University
[9] Kathy Roberson, Concordia University

As faculty, we should not be ashamed that we fail. Failure, after all, does serve a purpose.

> Charles Kettering, the namesake of my university, said lots of things about failure:
>
> "An inventor fails 999 times, and if he succeeds once, he's in. He treats his failures simply as practice shots."
>
> "99 percent of success is built on failure."
>
> "Every great improvement has come after repeated failures. Virtually nothing comes out right the first time. Failures, repeated failures, are fingerposts on the road to achievement."
>
> "The biggest job we have is to teach a newly hired employee how to fail intelligently. We have to train him to experiment over and over and to keep on trying and failing until he learns what will work."
>
> Kettering, by the way, had over 200 patents, including the patent for the electric car starter. I'm guessing he failed a lot.[10]

3. Lectures

Everyone makes mistakes, but a class will appreciate it if you promptly correct them. For example, if you fail to come up with the right answer to a question during class, you might e-mail it to the student or to the entire class shortly after the lecture. Sometimes the problem is broader—a class just doesn't understand a lecture, for example. In fact, they may even talk you into believing there is something wrong with it when there isn't. In this case, you might improve your explanation, put it on the Web, and discuss it during the next class.

Nowadays, many lectures involve some kind of computerized display. Try to avoid a single point of failure. For example, bring a laptop along in case you can't log in on the classroom computer. Put your presentation on a flash drive as well as on the fileserver. Christine Halsey[11] gives this checklist:

- Plan on extra handouts if visuals don't work.
- Install programs you need on more than one computer.
- Always carry extra copies of handouts, software or other files you may need (hard copies and digital copies)
- I email myself extremely important things in case I have to pull them up somewhere else.

One other respondent mentions that files should be saved in two formats (e.g., PowerPoint and PDF) just in case the presentation software does not work the way you expect. It is also a good practice to make sure you know how to change projector bulbs [1]. The same author also suggests having backup handouts, but often the amount of paper involved would be prohibitive. A better approach might be to carry originals, from which copies could be made on a nearby copier.

It's important to prepare well for class, as this anecdote illustrates.

> I just teach [part-time] on the side (over 10 years now). One of my first assignments way back

[10] Jim Huggins, Kettering University. Most of these quotations can be found at http://www.brainyquote.com/quotes/keywords/failure.html
[11] Christine Halsey, DeVry University

> when was teaching an environmental regulatory compliance class at a local CC (I'm a civil/environmental engineer). The students were not your typical 18-22 year olds, but older adults looking for a new career.
>
> About the 3rd or 4th lecture (it was a once-a-week, 3-hour marathon class), I got caught up in things at my real job and arrived relatively unprepared and tried to wing it. Bottom line, it didn't work and when I tried to apologize at the end (citing my work commitments), the students weren't buying it (saying, rightfully so, that they too had other commitments, but had made it there and were prepared).
>
> The next week, I showed up, told them they were right and proceeded to re-do the entire lecture. I also juggled the schedule (dropping some less-than-essential topics) to make space for the redo. Since then, I've never gone into a class unprepared (although I have had things out of my control go haywire - balky computers, overhead projectors, lab stuff, you name it, but the general principle in response is the same).[12]

If you do make a mistake, it's important to correct it quickly, before confusion has a chance to propagate:

> [In] a lecture I gave with PowerPoint on Friday[,] I somehow got myself confused, and I know I was getting the class confused. After class I went back to my office and looked my assembly code over more closely and it turns out that everything was correct with the slides (that I had used in last year's lecture). So I immediately made modifications/improvements to those slides in PowerPoint. Within a half hour after the class was over I had reposted the improved slides on Blackboard, emailed all of the students notifying them about the new posting, explaining why I wasn't "seeing" the code right, and apologized for the confusion I caused for them. On Monday, I put those slides back on the screen, and re-explained the code correctly, and with more confidence. The root cause of this, obviously, is not reviewing the slides well enough before giving the lecture.[13]

But, if you find yourself making the same mistake repeatedly, the students may be able to learn from it, as Jim Huggins explains:

> As a young faculty member teaching CS2 on a regular basis, I made the decision to develop the algorithms live on the chalkboard in front of the students, so that I would proceed slowly enough for the students to follow what I was doing. I found that I was making mistakes in the code, which I'd correct when pointed out by students.
>
> A couple of terms later, I discovered I was making the same mistakes over and over again, and resolved to learn the algorithms better. But then I realized that the mistakes I was making were meaningful. For example: I was forgetting when inserting into a linked list to remember that you need special-case code to handle inserting at the head of the list.
>
> So now, I deliberately make that mistake every time, and then ask my students to find the mistake, pointing out that there are special conditions like beginning/end/empty list that one should always consider.
>
> So, making a mistake becomes a teachable moment.

After each lecture, lab, or test, it is a good idea to make notes of what went right or wrong. Acquire the discipline of studying these notes carefully the next time you do the same lecture or lab, or use the same question on another test.

---

[12] Jeff Staudinger, Engineers Dedicated to a Better Tomorrow
[13] This author has requested anonymity.

Finally, rather than being embarrassed by students who know more than you do, make them members of the team:

> One more thing: I have a policy that if a student in my class knows more about the subject than I do, I invite him or her to prepare and present the lecture and to prepare the homework. I think it builds a level of trust between the students and me. This happens only rarely, but sometimes I have students in my class that are middle-management and they may very well know more about an area than I do.[14]

4. Homework

The best way to avoid problems in homework is to do the problems completely (or have your TA do them, if you have a TA). Even textbook problems are sometimes not solved in advance; if an answer does not appear in the solution manual, that may or may not be because no answer has ever been derived. Reusing problems from earlier semesters is a way around this problem, but be aware that the students may have access to the solutions [2, 3].

If errors are encountered in homework, it is often worthwhile to reward the students who find them.

> For courses that involve using software I developed, I give students a small amount of extra credit for any bugs they uncover and announce it ahead of time. I only reward the first person to uncover the bug - so I use it to motivate students to start early (and find any major problems early). Fortunately, I haven't had any major problems but a few students earned some extra credit finding bugs in interesting cases. There always seems to be a student or two that is really keen on trying to break the software.[15]

One instructor tells how he made a game of it:

> While grading the first homework assignment, I discovered that I'd made a mistake in the sample solutions that I handed out to students after they turned in the assignment. So I found a picture of a real ugly stag beetle and created a "wanted" poster. The first student to report a bug in any sample solution set, an assignment, or other course handout would receive a "bounty" of one extra credit percentage point on the cumulative course score (the curve was determined before factoring in the bounties). The student had to point out the bug and explain why it was a bug.
>
> The intent was two-fold. One, to keep me honest. Two, to encourage students to look at the sample solutions. There was an interesting side-effect. When I handed the students their exams, the first thing most of them did was look through the exam front to back, presumably looking for an ambiguity or some other bug that would win a bounty. But, it seemed to me that most of them then completed the problems in non-numeric order, presumably having identified which problems would require the least effort based on how each had studied.[16]

Sometimes it is possible to ask the students to fix the problem.

> In the first class that I taught as a new assistant professor in 1978 I had a takehome midterm question that I asked them to solve via dynamic programming. I realized a day or two after I had distributed it that the problem did not have the optimal subproblem property (the optimal solution to the entire problem is achieved by solving all subproblems optimally). The problem was

[14] Nancy Dorset, Penn State-Fayette
[15] Eric Larson, Seattle University
[16] Christopher Bohn, Air Force Institute of Technology

> changed to, "Why does the dynamic programming solution to this problem not give the correct answer?" Fortunately they still had several days to solve the new problem![17]

If the error is discovered later, the due date can be extended. But this may get in the way of an orderly progression of material, or damage students' confidence in their ability to do the homework.

> Postponement is awful because it kills any hope for a sustained, balanced work level. …
>
> But even if it is possible to introduce the missing material, academic damage has been done: the tacit knowledge of the student when considering homework that "she/he must have been taught all the necessary skills and information" is shattered. It is that tacit knowledge that makes the difference between an academic homework assignment and a genuine research project.[18]

However, once students are sufficiently mature, they should be able to deal with changing requirements:

> For an advanced … class, tell them this is what it is what it is like out in industry. … You get an assignment you think will be easy and you find out the person before you did not do their job correctly. Your job is to get it working with the changes that were supposed to be initially made. Depending on how much change, I would either change the deadline or keep it the same. If they request an extension for the deadline (I won't give it unless they request it), they must give me a sound, logical reason for it and support their request. Again, out in industry, you will not be able to get what you want unless you ask for it and can convince people that it is the right path to go.[19]

Another approach is to decrease the weight of the buggy homework. But this has its disadvantages as well:

> [M]y suggestion is to adjust the weight of the item (or curve the exam/project). I really don't like doing this because if students assume that exams will be curved then if *all* students don't study or do the assignments, and *all* students perform miserably on the exam/project, then the curve will support their laziness. I guess that, over time, the quality of a faculty member's exams/projects improves and becomes more realistic. Each iteration, of course, adds or replaces new content and some of these questions may be bad. These few questions can be adjusted for (i.e., many mass produced exams will frequently add new questions which are not graded for the student taking the test but used only statistically to indicate the goodness/badness of the question).[20]

If the deficiency is limited to a particular problem on a problem set, then that problem can simply be disregarded in grading. Some instructors will spend time reviewing the problem that was dropped, and then use it on a subsequent exam.

5. Labs

Labs have a lot in common with homework, but they are different because the instructor is normally present. If you notice mistakes in the handout during a lab, it is a good practice to make corrections immediately, while the errors are still fresh in your mind, so that the same error will not occur the next time the lab is assigned. The changes should

[17] This author has requested anonymity
[18] This author, too, has requested anonymity
[19] Amber Hurst, Wake Tech
[20] Bruce Barton, Suffolk Co. Community College

be made in the wordprocessor file, either corrections, or notations in red of what needs to be corrected.

If students are obtaining very different results from what you expect, it is good practice to redo it yourself or have a lab assistant redo it. If the results indicate that the students were wrong, then you can explain the error at the next session, but give partial credit for the results.

One of the instructors told of a situation in which his students were doing an introductory lab on op-amps. He had been told that the breadboards were shielded, but quickly found out that this was not the case. He tried many approaches to ground them, but none succeeded; he couldn't eliminate enough noise. Instead of continuing the lab, he gave a spontaneous lecture on op-amps, and showed them why their experiments were being overwhelmed by interference.

To complete the lab, he gave the students two options. First, they could redo the lab at a later time with proto-boards, which were shielded. Alternatively, they could write up a proposal to perform the experiment, and outline expected measurements, including charts and graphs to support their reasoning.

The students in the class were mostly experienced electronic technicians. All of them chose the second option, which resembled situations they had encountered in their work experience. The instructor redid the experiment with a proto-board, and demonstrated it to the class at the beginning of the next week's lab.

One instructor in a linear electronics lab tells that students will frequently burn up a device or component. When that happens, they present a "show and smell" session, in which they explain to the rest of the class what they did wrong, and allow the others to inspect and smell the device. This helps the other students to see what *not* to do and helps protect against repeat accidents and the resulting losses.

Sometimes, when a lab just doesn't seem to be working out, it makes sense just to cancel it. Another instructor relates,

> I could see that I had gotten the students into a lab to develop microcontroller software that was just too much for a 3-hour lab, and I don't think it would have worked out well no matter how long we worked on it. So I just asked the students, "Do you think it would be OK to just smile, and wave goodbye to this lab?" They all agreed, and we waved goodbye with a smile.[21]

5. Tests and Exams

No other topic received as many comments as tests and exams. First, there was advice on how to prevent embarrassing moments. Most of it centered around working exam questions carefully before giving them to students. One instructor writes,

> Honestly, I've never had this problem happen to me because I take every test I give my students. I give them a 5-1 ratio. If I can't finish a test for a 50 minute class in 10 minutes, I shave off questions. The better students finish the test in about 30 minutes, and usually even the slowest

[21] This author has requested anonymity.

students are pretty much done at the end of class time. It also forces me to really focus on what I want to test them over without asking redundant questions.[22]

If you solve an exam ahead of time, you can also pass out solutions to students as they leave. Wayne Wells[23] tells of one instructor who did this:

> [A]s each student completed the work and turned it in, she handed them a copy of the solutions with the rule that they could not look at it or utter a word until they had left the test room. They were free to celebrate or beat their heads against the wall out there.

Writing up the answers in advance also gives you the opportunity to edit them as students ask questions during the exam. You can adapt your answers on the spot, or later when you grade the tests. Not only does this explain to the students why certain answers are wrong, it also helps you to revise the question to prevent misunderstanding, if you ever use the question on a later exam.

However, writing answers in advance does not prevent all errors. You need to be careful during grading, lest you penalize students for a correct answer that does not agree with yours. Mark Hoffman[24] explains,

> When I first grade a newly-created exam or question on an exam, I double check students' answers against my own. I may have made an error in calculating the answer, or I may have made an assumption in constructing the question. Either way, I make sure an incorrect answer it truly incorrect. I regularly suggest to students that they list assumptions or ask the instructor (if feasible) for clarification if they are unsure about the question.

There is a pitfall in allowing TAs to grade exams: They may assume that the instructor's answers are correct, no matter how many students get a different answer. I find it necessary to grade the first few exams with my TAs before they can do a consistently good job on their own.

If a bad question does make it through to an exam, the easiest thing to do is to adjust the weight. You can give the students full credit for the question, or perhaps give them credit if (and only if) they put some answer down on the paper. You can disregard the question and calculate a grade based only on the other questions. You can also do this if a multipart question has a bad *part*. When I inadvertently asked one part of a four-part question over a lecture I said wouldn't be covered on the exam, I disregarded that part and multiplied the score for the other parts by 4/3.

If the average on the exam is low, you might consider dropping questions that everyone got wrong. If most students got a question wrong, you might disregard it, but give extra credit to anyone who got it right.

You might instead curve the grades to produce an acceptable median (not an acceptable mean). Or, you can weight the whole test less heavily in calculating a final grade. But this is not to be done lightly, because it violates an implicit contract with the class and penalizes students who have done well, relatively speaking, on that exam. Jim Huggins

[22] Misti Clark, Ouachita Baptist University
[23] Wayne Wells, U. of Texas-Brownsville
[24] Mark Hoffman, Quinnipiac University

explains:

> Students want a fair grade. They want to know that their grade in the course is an accurate evaluation of how well they learned the material of the course. This leads to all sorts of interesting implications which we usually provide: full disclosure of how our grading system works; evidence that the system will be applied consistently, without bias or prejudice; evidence that the evaluative tools used will be relevant to the course material, and so on.
>
> In the scenarios [described in this paper], the instructor has attempted to provide all of the above, and in one instance has failed. One is left with a couple of bad choices:
>
> - stick with your pre-announced system, which now includes an unfair evaluation datapoint
> - alter your pre-announced system to exclude the unfair point, at the cost of violating your announced rules
>
> If we keep in mind that the primary objective is supposed to be providing a fair grade, it seems to me that the latter choice is the obvious one. Providing a fair grade is more important that slavish devotion to a pre-announced system which is no longer relevant.
>
> Obviously, this has to be done with care. But, on the whole, I think students would agree that, of the bad alternatives, changing the grading system makes more sense.
>
> [Aside: this is a little easier to do if you have a grading system which doesn't work on an absolute scale. If you're going to be scaling/curving the results anyways, it's not that much more difficult to exclude a couple of bad datapoints as part of the overall adjustments.]

When an exam is too long, you can "curve" it, as described in Section 2. But you can also drop questions, turning it into a "do *k* of *n* questions exam," and raising the weight of each question proportionately so that the points still add to 100%. If a student has done more than *k* questions, you should count the student's *k* best answers.

Another approach is not to do anything for the whole class, but rather to defer individual decisions until the end of the term. Eric Larson describes what he does:

> If there was a problem with a test or an assignment, and a student thinks they were treated unfairly as a result, I usually tell the student that I will put an asterisk in the gradebook by that score and that if that score was an outlier with respect to the rest of the graded activities, I will give them the benefit of the doubt when it comes to the final grade (I generally do this anyway when computing final grades). I have only done this three or four times and in all of those cases, the score was not an outlier and the grade was not changed.

All of the techniques described so far just adjust the scores for work already done. A poor score on an exam may also be indicative of poor knowledge of the material. In this case, it may make sense to give a homework assignment over the same material. Or, you might have the student retake the test for partial credit (i.e., to gain back a portion of the points lost on the test the first time it was taken).

5. Conclusion

When something doesn't work in class, a savvy instructor has many options. The main rule is not to make a decision that would unfairly penalize any group of students. It is imperative to keep track of what worked and what didn't, so you can learn from your experience. One of the instructors who responded said that his department had a regular post-mortem session after each semester in which they write out their recollections of what worked and what didn't, and their impressions of the group of students (because if an assignment failed, it might have been because the students were weak). But it's not necessary to wait until the end of a semester. Keeping a journal for each class day and each assignment helps you to improve the course the next time around.

Bibliography

[1] Cannon, Robert, and Newble, David, *A Handbook for Teachers in Universities and Colleges: A Guide to Improving Teaching Methods*, 4th ed., Kogan Page, 2000.

[2] Gehringer, Edward F., "Reuse of homework and test questions: When, why, and how to maintain security?" Frontiers in Education 2004, Savannah, Ga., October 23, 2004.

[3] Widmann, James, "Student use of author's textbook solution manuals: Effect on student learning of mechanics fundamentals," ASEE Annual Conference and Exposition, 2007.

EDWARD F. GEHRINGER
Edward Gehringer is an associate professor in the Department of Computer Science and the Department of Electrical and Computer Engineering at North Carolina State University. He has been a frequent presenter at education-based workshops in the areas of computer architecture and object-oriented systems. His research interests include architectural support for memory management, garbage collection, and computer-supported collaborative work. He received a B.S. from the University of Detroit(-Mercy) in 1972, a B.A. from Wayne State University, also in 1972, and the Ph.D. from Purdue University in 1979.