



---

## A Constructivist Approach to Communication Skills Instruction in Computer Science

Paul Gruba<sup>1</sup> and Harald Søndergaard<sup>2</sup>

<sup>1</sup>Centre for Communication Skills and English as a Second Language, and <sup>2</sup>Department of Computer Science and Software Engineering, University of Melbourne, Vic., Australia

---

### ABSTRACT

From the social constructivist perspective of education, learning is best achieved when students face complex, real world problems in which there are no clear answers. Faced with a sizable common goal, students work collaboratively towards outcomes and maintain ownership over key decisions. The role of staff is that of facilitators whose role is to challenge learners to explore multiple aspects of the problem as they go about reaching viable solutions. Such a role contrasts, for example, to an approach which sets out to lead students to a presumed correct solution that is already possessed by the instructor.

Based on these principles we designed and implemented a course on communication skills in computer science. Here, we describe our experiences using a student-run conference as a means to teach communication skills. In this approach, students were charged with the task of planning and organising a conference, including peer review, publicity, budget, sponsorship, web design, conference program, presentation schedule, speaker support, and catering. We describe the principles and their implementation and reflect on the outcome.

---

### INTRODUCTION

*Picture this:* In the first 10 minutes of our communication skills course, we announced to students that they were to hold a public conference 11 weeks later, and that they had to organise the event themselves. Our role, we told them, was to help facilitate their efforts. The first reaction from the students was silence, and then the questions began. Who was to fund the conference? How could we fit 65 student papers into one day? What *was* a conference? We began answering the questions, one at a time, and by the end of the hour we

---

Correspondence: Paul Gruba, Centre for Communication Skills and English as a Second Language, University of Melbourne, Melbourne, Vic. 3010, Australia. Tel: +61 3 8344 8991, Fax: +61 3 9349 3501, E-mail: p.gruba@animelb.edu.au

ourselves were wondering if we had made the right decision. No, we told our students, we did not have all the answers; yes, the event could turn out a failure. One point we made clear to the students: we told them that direct experience is the best way to learn about effective communication skills.

As it turned out, the conference ran as scheduled 11 weeks later and it was a clear success. The student presentations were professional; and the guest speakers were engaging the audience, many from outside the university, were lively and interested; and the food was delicious. As we basked in the glory of being the course conveners, we had to remind ourselves that all of this had been organised by our students. We had only provided a framework and they had filled it. From setting an attractive conference theme to getting financial sponsorship to writing the academic papers, it had truly been a student-driven event.

The aim of this paper is to describe what we and the students did that led to a successful conference. We first set out some of the pedagogical theory that motivated us. Next, we present the details of the course including the syllabus, the structure of student teams, and the assessment tasks. We then describe the conference in more detail and highlight the problems and successes that we encountered. We discuss a recent variation of the model. The paper ends with a discussion and suggestions for further development.

## CHALLENGES FOR COMMUNICATION SKILLS

The list of topics considered important in computing-related degree programs seems to be ever increasing. The computer science curriculum is in constant flux, trying to extend itself to cover the latest advances in computing technology, to incorporate new tools and techniques, and to develop new skills in students, in subject areas that often had no meaning a few years earlier. The challenge in curriculum development is to identify the general principles worth teaching, to identify and present to students the essential core of a new topic, and to integrate new with old in a seamless way.

In spite of the rapid changes, there are also constants in Computer Science education. Apart from teaching necessary technical skills, the aim of a complete computer science education is to produce individuals who are critical and constructive, and who can do research and work independently or together with a variety of people. Good communication skills are a key prerequisite for a successful professional life. The abilities to think critically,

to negotiate a point of view, and to reflect on one's own learning, are important skills that every student should develop before leaving university.

In response to a number of issues related to communication skills instruction throughout the campus, the University of Melbourne has established a 'communication across the curriculum' project, based on the success of several overseas initiatives in tertiary education (Bazerman & Russell, 1994; Young & Fulwiler, 1986). Project members work closely with academics in departments to help them foster the integration of communication skills instruction in their own disciplines. The project described in this paper resulted when the Department of Computer Science and Software Engineering requested assistance in developing the computer science communication skills course.

By necessity, most Computer Science courses include a considerable amount of practical project work. Projects aim not only at developing technical skills, but also at providing students with research, team-working, and organisational skills. In many courses, these aims are implicit. In other courses, especially in Software Engineering, they are particularly important and usually expressed clearly through the syllabus.

For communication skills instruction, one cannot hope to teach good writing or speaking systematically through a small number of lectures. Communication skills are obtained only through active engagement with the subject matter.

Consider writing skills. Good writers primarily learn their skills in their own time, and they succeed because they have learned to recognise good writing. They develop their skills because they assess writing critically, and, in particular, can improve their own writing through self-assessment.

Yet this sort of critical thinking is not easily taught, except perhaps through many and varied examples. This leads to the conclusion that our best tools in a short communication skills course are *examples* and *feed-back*, exposing students to a variety of writing and speaking, challenging their critical thinking, and providing them with ample comments on their own output. The challenge then becomes how best to do this with limited staff resources.

Communication skills courses are considered increasingly important, and yet, paradoxically, are in constant danger of being squeezed out of a more and more crowded curriculum (McGregor, Saunders, Fry & Taylor, 2000). Its integration into other project activity is a sensible approach, both as a response to the crowded curriculum and for reasons discussed in the next section.

A traditional approach to communication skills instruction may contain a mixture of lectures on theoretical and practical matters. On the practical side,

presentation technology may be surveyed and document-processing tools explained. On the theoretical side, elements of communication may be set out. Senders, messages, channels, receivers and feedback are each given a role and a definition. Competing theories may be surveyed; 'transmission' models of communication (Berlo, 1960) may get dismissed in favour of more recent 'transactional' or 'meaning-based' versions (Barnlund, 1970). Throughout lectures, the highly contextualised nature of communication is emphasised. In traditional approaches, theory is rarely put into practice. Classes proceed with application tasks, case studies, or discussion of communication practices. Students and staff alike often complete these classes wondering if they are just pretending to communicate. Real communication, it appears, takes place somewhere else.

Aware of these problems and frustrations, we searched for an innovative approach to make communication skills clearly relevant to Computer Science students.

### CONSTRUCTIVIST PRINCIPLES

Social constructivist theories, widely used in mathematics instruction, provided us with inspiration (Steffe & Gale, 1995). Driscoll (1994), an educational psychologist, has provided a clear roadmap in her articulation of 'social constructivist' principles of learning. The social constructivist view, applied to educational settings, has served as the basis for other learning innovations (Duffy & Jonassen, 1992).

To integrate communication skills, we directed students from the start to develop, administer and realise a formal academic conference. The 'conference approach' fits well into constructivist approaches. Five principles set out by Driscoll (1994) provided us with a solid framework to build the following.

#### **Integrate Authentic Activity within a Complex Learning Environment**

The realisation of an academic conference is an authentic task within a tertiary setting. Students readily identify with the project and feel a sense of ownership towards it. Importantly, working through processes to organise a conference allows for practice of communication skills interpersonally (within small-committee work), institutionally (as an entity within the university) and personally when they present papers. At each juncture, discussion of theory precedes the upcoming activity.

**Emphasise Social Negotiation as Integral to Learning**

Placing a group project at the core of the class focuses student attention on group dynamics and interpersonal communication. The success of the conference hinges on students' collaboration. Social negotiation, both within and outside the class, must be used to organise the variety of necessary tasks. Our role as instructors is to provide for reflection on the social processes as they take place.

**Juxtapose Content, Include Multiple Modes of Representation**

Effective communication, by its very nature, contains a juxtaposition of ideas and information that the students struggle with from day one. The nature of the group project and, importantly, the assessment tasks which build up to and include the conference, results in the students becoming familiar with and experienced in presenting their work in a variety of modes: spoken, written, individual, group, and electronically-assisted.

**Keep Instruction Relevant to Student Needs**

Despite our provision of structure through the conference approach, where the students are given the basic task in advance, there is great scope for students to shape the conference. Their ownership of the event is made clear from the beginning. Students set the theme for the conference, they find appropriate target audiences, they find guest speakers and arrange for venues, catering and security. Our role as instructors is to facilitate the project, not to intervene. As a need arises in relation to the conference — and this includes conflicts within the group — we provide instruction and perspective. Students then apply their understanding directly and, we hope, another aspect of communication skills is deeply learned.

**Reflect on Practice**

As Winn (1991) notes, there is a point where the complexity of learning makes prediction of performance and prescription of instruction impossible. To monitor that complexity, we utilise a series of assessment tasks throughout the semester.

1. In the beginning, we ask students to define their tasks more clearly through 'committee statements'. Each committee writes a mission statement, a work-plan with time-lines, and a progress report identifying achievements so far. It also lists its members and their duties.

2. Drafts of students' papers are distributed and reviewed by students and staff.
3. As an exercise in self-assessment (and self-promotion) students produce a statement outlining their contributions to the event.
4. At the end of the course, an in-class essay provides an opportunity for students to reflect on their learning. We find that such post-project reflection encourages critical perspective and provides an important forum to discuss theory against practice.

### COURSE CONTENT

As we planned the course we took the principles of social constructivism seriously. The main challenge was to enable students to have time for the preparation of a conference within a one-semester course. We had to organise the lectures so that material was presented in a timely manner and make sure that lectures and tutorials were synchronised with the schedule of deadlines for assessment tasks and peer review.

In 1998, the class consisted of mostly Bachelor of Computer Science and Bachelor of Engineering (Software Engineering) students. There were 65 students enrolled in the class. The majority were third year students, and 20 students were in their fourth year. A sizable portion of the class were software engineering students and thus came to the course with strong teamwork experience. The students came from a variety of backgrounds and we estimated that more than half were non-native speakers of English. Most of the students intended to work strictly in computer science. Few had taken dedicated humanities classes.

There was one weekly lecture and one weekly tutorial/lab class. We shared the teaching evenly. Students were told that the objectives of the course were to

- improve oral and written communication skills, and in particular to be able to write for a varied readership;
- become familiar with document processing tools and presentation tools that are commonly used in the discipline;
- become able to use research libraries and other resources efficiently;
- gain some experience in collaborative work, including organising meetings, co-authoring, and working in teams on problem-driven projects.

Table 1. Deadlines.

Date	Activity	To whom
26 March	Abstracts (ASCII) submitted	Steering Committee
2 April	Abstracts refereed	Steering Committee
10 April	Committee statements (LaTeX)	Lecturers
27 April	Papers (LaTeX)	Steering Committee
4 May	Papers refereed	Steering Committee
11 May	Camera-ready versions (LaTeX)	Steering Committee
16 May	Conference	
22 May	Conclusions, thank-yous	Relevant persons
26 May	Self-assessment	Lecturers

The following topics were covered in lectures and tutorials: small group dynamics; library and Internet resources, personal information management; document preparation with LaTeX, bibtex and related tools; writing an abstract; drafting; cohesive writing; critical thinking; academic writing; citation; assessment and peer-refereeing; giving a talk; presentation technology.

The major challenge was to schedule these topics so that they would support student activities when needed. A reading package was prepared as support for lectures, with excerpts from various texts (Goossens, Mittelbach & Samarin, 1994; Lampion, 1994; Macnamara, 1996; Mohan, McGregor & Strano, 1997; Peyton Jones, Hughes & Launchbury, 1993; van Leunen, 1992; Zobel, 1997) together with our own teaching material. Table 1 outlines the plan for lectures and tutorials.

## ASSESSMENT TASKS AND ACTIVITIES

In advance of the first lecture, we set out six committees and randomly assigned students to each. We announced these assignments near the end of the first lecture, and told the class to choose one person from each committee to serve on a seventh 'Steering Committee'. The committees and their proposed tasks were the following.

### **Finance Committee**

Set budgets; monitor spending; maintain appropriate accountancy procedures; distribute monies; finalise accounts.

**Advertising Committee**

Produce publicity materials; contact journalists; liaise with identified audience groups; produce posters; write pre- and post-conference press releases.

**Sponsorship Committee**

Identify potential sponsors; develop sponsor packages; contact sponsors; ensure sponsor hospitality; thank sponsors.

**Event Committee**

Manage project morale; choose and reserve venue; organise social events; coordinate catering; provide on-site assistance; administer registrations; post-signage; assist presenters; supervise clean-up.

**Publications Committee**

Construct web site; establish paper guidelines; schedule peer assessment; collect papers; produce conference booklet; publish papers.

**Program Committee**

Identify potential invited speakers; invite speakers; prepare speakers; create presentation schedule; ensure speaker hospitality; send letters of thanks.

**Steering Committee**

Liaise with academic staff; liaise with and coordinate committees; lead and solve problems; produce executive summary reports; set priorities; manage milestones; establish conference theme.

To integrate all of the 65 students into committee work, most committees consisted of 10–12 members. In recognition of the relative workload, the Finance and Sponsorship Committees had six members each. The Steering Committee consisted of five elected members, drawn from the other teams.

The assessment in the course had five components.

**Committee Statement (15%)**

The committee statement was a group-produced document written in LaTeX that included the committee's mission statement, individual member profiles and responsibilities, and a work-plan that set out project milestones. The document was assessed for completeness, evidence of group coherence (for example, consistent tone of member profiles), structure, style and grammar.

**Conference Paper (40%)**

The conference paper could be produced individually or collaboratively, with up to three authors per paper. It had to be between 2000 and 6000 words in length. It was made clear that a recognition of audience awareness and readership was critical to the success of the paper — it should be written for someone outside the discipline. Typically it would explain and discuss some non-trivial issue in computer science to the generalist in the academic community or to an interested and well-motivated lay person.

**Conference Presentation (15%)**

This was an oral presentation of the conference paper. All students had to present. A dozen of faculty from the Department of Computer Science attended the conference, and they were given the job of evaluating allocated talks based on a standard guideline. Each of the student presentations was assessed by three evaluators in the categories of content, preparation, and delivery.

**Self-Evaluation (20%)**

The self-evaluation was a formal one or two-page memo addressed to the Head of Department. It described the student's contribution to the class project. This was intended also as an exercise in producing material for self-promotion.

**In-class Essay (10%)**

At the end of the semester, following the conference, the in-class essay was intended as an opportunity for students to reflect on the role of communication skills theory in actual practice. Students were asked to choose one of the following topics.

1. This course used a 'conference approach' as a way to teach communication skills. Choose one aspect of the conference project (organising, publicity, theory), assess it critically and suggest ways to improve the approach.
2. In brief, communication theorists argue that small groups go through five predictable stages of development. Against your own experiences gained in committee work, assess whether or not these five stages actually exist.
3. Based on your own experiences in this course, argue for or against the validity of the transmission model of communication.
4. To give a truly effective oral presentation, would it be better to deeply know your subject matter or deeply know your audience?

Table 2. Lectures and Tutorial Plans.

Date	Lecture	Tutorial/ lab	Reading (from package)
3 March	Introduction, committee formation, Q&A	No formal session	
10 March	Small group dynamics	Initial committee meetings	Mohan et al.
17 March	Academic writing; Writing an abstract	Introduction to research	Zobel; Gruba
24 March	Document preparation; LaTeX basics	LaTeX	Lamport, Goossens et al.
31 March	LaTeX: Formulas, tables and figures	LaTeX	
7 April	LaTeX, bibtex and related tools	LaTeX, xfig, jgraph	
21 April	Critical thinking; drafting; cohesive writing	Discussion	van Leunen
28 April	Assessment, critique, peer review	Critical thinking exercise	Gruba
5 May	Giving a talk		Peyton Jones et al.; Macnamara
12 May	Presentation technology	Presentation practice	Levin and Redell
19 May	Reflection and feedback	Reflection	
26 May	In-class essay		

The first and the last topic proved most popular. Table 2 shows the tight schedule of deadlines that we had to impose for the assessment components.

### STUDENT ACTIONS AND REACTIONS — HOW IT WENT

From Week 1, the students had been placed in committees. Their first meetings were tentative and hesitant. Students were feeling their way with each other and with us, trying to create a solid idea of what was expected.

We were not always able to advise them, but could only remind them that sometimes there is no clear ‘right’ answer. Often, decisions have to be made with just the information given. Holding to our constructivist perspective, we decided to let the students struggle. It was not always easy to watch the students get frustrated, but there was definitely a strong sense that communication skills were being practiced.

After getting to know each other, students did indeed begin to communicate. We worked alongside the members of the Steering Committee most closely. At first, members of the Steering Committee established their boundaries and tasks, and this filtered down to the other committees and eventually to each individual. A period of confusion followed and lasted for a couple of weeks. This was sometimes unsettling and there were spirited discussions. The conference theme, and whether a small attendance fee should be charged, were key topics. Eventually, through the discussions a theme emerged: “*A World Going Digital: Computer Science and Its Impact.*” Students chose this theme because it was broad enough to allow for a variety of interests, and because they felt that such a theme would also appeal to a wide audience of fellow students from other disciplines.

The other committees then set about their own set of tasks. Often they had to go beyond the limited set of tasks that we had initially provided; we had only given them a framework. Students came to realise that they had complete ownership of the up-coming event. With this realisation came an outburst of student energy and creativity that took us by surprise.

The Sponsorship Committee soon learned to approach graduate recruitment officers in large corporations, rather than some unknown secretary. Those in charge of recruitment usually have a budget for exposure to students. In 6 weeks, the committee secured nearly AUD 3000 for the conference from private and university sources. The Finance Committee produced worst-case and best-case budgets for the conference and allocated money accordingly. Other committees produced stunning conference web pages, an attractive poster for advertising the conference to other students, and promos for the local community radio station.

The carefully designed program booklet listed a total of 46 papers for the 65 students because, like an academic conference, some of the work had been written by multiple authors. The Program Committee had grouped the papers into sessions on the Internet, e-commerce, software engineering, computer perception, education and social impact, future technologies, artificial intelligence, and games and entertainment. Spread out over three separate lecture theatres, the program was accomplished within 5 hours of presentation. Nearly all of the presentations were supported with visual materials or were delivered with electronic aids.

Two invited speakers were featured. In the afternoon, the Department hosted a demonstration of two robots. In all, there were approximately 130 attendees of which half were students. A total of 10 or so staff members from

the department attended, and another 12 academics from outside the department came along. Postgraduate students, members of the general public and family and friends made up the rest of the audience. Fortunately, there was enough catered food to feed everyone.

### PROBLEMS AND SUCCESSES

Given the somewhat ‘high risk’ approach that mounting a student conference entails, it would be folly to say that our attempt at innovation was not without some problems. We encouraged ongoing student feedback and evaluation and stayed flexible enough to make changes. One key concern about the conference project was the somewhat uneven distribution of student workloads. Several students approached us about this problem; our advice to students included tips on better time management, increased team work and a general call for them not to be too obsessed with the conference. We noticed that conscientious students would work long hours to assure that particular details were attended to and carried out properly. Of course, without their dedication, the event would not have been as successful or run so smoothly. In private discussions, these students said that they were working particularly hard because they did not think it was easy to share the task, some of their team members had let them down, or they felt particularly concerned about doing a proper job. Our efforts to encourage the sharing of tasks — either through individual lobbying or class announcements — was met with limited success.

Another point that was raised repeatedly by the students was the challenge of adopting to an unfamiliar style of learning. Clearly, our approach was a departure from the usual lecture and tutorial pattern of instruction. More importantly, the lack of a strong set of parameters concerning, for example, how to accomplish tasks in self-managed teams was regarded as a challenge. As instructors, we often debated whether it was necessary to intervene when students floundered and, if so, how much to intervene. While holding to the non-interventionist ideals of social constructivism, we nonetheless struggled with the real possibility of spectacular failure.

A third problem we encountered was the workload the project put on us. Although, we had expected some increase in the amount of effort it would take to develop a new course, we were overwhelmed at the quantity of details that required our direct attention. Of these, interactions with students who were contacting sponsors, keynote speakers and financial matters seemed to

demand the bulk of our time. We regretted the investment of this time to matters that, on the surface, had little to do with the aim of the course, that is, with teaching communication skills.

The conference approach also fostered many benefits. Many students felt a greater sense of camaraderie and got to know each other much better. The randomly composed small committees working towards a central project gave students a chance to create new networks.

Students generally came away from the project with a number of skills that could transfer to other situations. Software Engineering students were less appreciative of the course as an exercise in teamwork and organisation, as they felt that these aspects are already well covered in their degree program. They would have preferred more time for research and writing.

Student feedback improved with time. After the conference, there was clearly a great deal of pride over the way it had turned out. A large number of students felt that the conference was so successful and of such importance that they would include their contributions in their resume. Some of the most encouraging feedback has come from a student who wrote to us after he had joined the workforce, to confirm the value placed on good presentation skills in his workplace.

A positive aspect of the conference was its ability to reach out to the larger community. In addition to getting recognition of their value from sponsors, students saw value in speaking to members of the general public about current issues in computer science. At our event, attendees ranged from teenagers to pensioners, and the best questions and comments, as well as the strongest support for the speakers, usually came from this part of the audience. Staff from the department were particularly pleased to see the non-academics come to what otherwise might have been a fairly constrained event.

### VARIATIONS OF THE MODEL

We have continued with the conference model in spite of growing logistical problems. Our course is now merged with a course on 'Professional Issues' which covers topics such as ethics, contracts and projects, intellectual property, privacy, censorship, security, electronic commerce, and information technology politics. The merger makes good sense, as students prefer to examine these type of topics in their communication skills papers.

Following the merger, however, we no longer have the luxury of focussing our sessions solely on improving student communication skills. The com-

munication skills sessions have been integrated into the overall lecture stream. Although, Gruba remains responsible for the communication skills part, the number of student enrolments has more than doubled and the teaching staff have changed.

So is it still feasible to run a conference? Does it make sense that we stick with our adopted constructivist principle? Clearly we have noticed that the model has become strained and there are considerable logistical challenges now that student numbers exceed 150.

Despite the changes in the course, we believe that it is still worth helping students organise a conference. These days, the entire conference is organised and run by a group of 18 students. During the first lecture, the entire class is introduced to the idea of the conference, and students who are interested to organise the event 'self-select' to be part of the conference organising team. The other students write the papers for the conference. Gruba facilitates the project management in eight 1 hour tutorial sessions.

Materials and experience from the 1998 event serve as handy guidelines and speed up development. Students make good use of templates from previous years. For example, many of the forms that earlier students developed can now be used with minor changes and lists of potential sponsors can be shared. The smaller group certainly makes decisions faster. We realise, of course, that the students who have volunteered for the project are the ones who are very well motivated.

Ironically, even though the 'inefficiencies' of our first student conference project have been reduced, such an approach has also weakened our resolve to hold onto key constructivist principles. Ownership of the project, for example, has now been lowered because the project appears to be no more than a departmental tradition; during organisation, social negotiation has been minimised as the students seek to emulate the success of previous efforts. Nonetheless, the student organisers set the conference theme, hold meetings, and improve their communication skills as they work to achieve an ambitious and an ill-defined goal. The conference itself now requires four lecture theatres to hold parallel sessions and continues to be a great success with both speakers and audience.

## CONCLUSION

We know of relatively few reports of similar experiments in communication skills instruction. Gruba and Lynch (1997) describe an earlier version of the

conference approach in other disciplines and we have presented our work at professional forums (Gruba & Søndergaard, 2000). Börstler and Johansson (1998) report on a one-semester course very similar to ours, with a final public conference, but with a much smaller class, less student involvement in the organisation, and without the assessment components we introduced for the purpose of nurturing reflection. On the other hand, their students are expected to produce more substantial papers, with a stronger focus on research. Their experience generally agrees with ours. The high workload and the short time frame are identified as major problems. They end up opting for a longer course so as to include more lectures and to leave time for a peer-review process.

We are still wrestling with matters to do with the ‘appropriate structuredness’ in the approach. To what extent should we guide confident students and to what extent do we let them undergo a period of trial and error? Ideally, we should be able to ‘scaffold’ the tasks during the course of the semester so that the skills and lessons learned 1 week build on the work of the next week. In reality, it is not that simple. Students come to the class with differing levels of both communication skills and organisation experience; we ourselves have been either frustrated or taken aback at the pace of decision-making on important matters. Issues that we had initially thought trivial became large; notions we ourselves had held dear were readily dismissed. The ebb and flow of such a class, and every particular group of students, is difficult to gauge. That makes it difficult to manage and plan for.

We also need to better define resource and task allocation. We do not want to foster a situation where one team’s task is seen as being easier than other teams’ tasks. On the other hand, we do not want to make it seem that all tasks are equally difficult and require the same effort. In any case, we have no formula to allocate equal workloads to students. Too much depends on how each individual perceives what they need to do, (for example, sandwiches for the catered lunch or a hot catered meal), the level of quality they want to achieve, (for example, a photocopied program guide versus a full-colour glossy brochure) and how each team delegates responsibilities amongst its members.

So is a student conference an effective way to teach communication skills to computer science students? One student remarked that he had learned little about communication during the semester and found the approach better suited to teaching management or organisational skills. Although, we acknowledge such criticism, we support the idea of a conference for several reasons.

1. Skills are developed in the individual, small group and public domains of communication.
2. The 'ill-defined' nature of the project resonates with real world situations.
3. The conference itself is an ordeal for some students, but there is a clear feeling of pride amongst students after a job well done.
4. Communication takes place for real purposes and real audiences, as opposed to the style of communication that occurs in small student, only tutorial sessions.
5. The conference approach with its peer-review harnesses an important resource, namely peer-tutoring, and encourages constructive critique amongst students.
6. Staff in the department are now more sensitised to issues of communication skills development in student life.

Finally, the success of the conference must also be seen in the context of the excellent resources that have been available to our students and ourselves: Good venues and presentation tools in all theatres, good technical support, and strong support from the Department of Computer Science and Software Engineering.

#### ACKNOWLEDGEMENTS

We would like to thank Chi-Keen Low for providing tutorial support to students, general support for us, and excellent web-based tools for peer-review management. We also thank Les Kitchen for help with robot demonstrations and the staff of the Department of Computer Science and Software Engineering for their assistance with assessment. Finally we congratulate our Communication Skills students with their highly successful conferences and thank them for accepting our challenge with such enthusiasm.

#### REFERENCES

- Barnlund, D.C. (1970). A transactional model of communication. In K. Sereno & C. Mortensen, C. (Eds.), *Foundations of communication theory* (pp. 23–45). New York: Harper and Row.
- Bazerman, C., & Russell, D.R. (Eds.). (1994). *Landmark essays on writing across the curriculum*. Davis, CA: Hermagoras Press.
- Berlo, D.K. (1960). *The process of communication*. New York: Holt, Rinehart and Winston.

- Börstler, J., & Johansson, O. (1998). The students conference — A tool for the teaching of research, writing, and presentation skills. *SIGCSE Bulletin*, 30, 28–31.
- Driscoll, M.P. (1994). *Psychology of learning for instruction*. Boston: Allyn and Bacon.
- Duffy, T., & Jonassen, D. (Eds.). (1992). *Constructivism and the technology of instruction: A conversation*. Hillsdale, NJ: Lawrence Erlbaum.
- Goossens, M., Mittelbach, F., & Samarin, A. (1994). *The LaTeX companion*. Reading, MA: Addison-Wesley.
- Gruba, P., & Lynch, B. (1997). Constructivist approaches to communication skills instruction. In R. Kevill, R. Oliver, & R. Phillips (Eds.), *What Works and Why? Proceedings of ASCILITE'97; The Australian Society for Computers in Learning in Tertiary Education* (pp. 245–249). Perth: Curtin University of Technology.
- Gruba, P., & Søndergaard, H. (2000). Transforming communication skills instruction: The conference approach. In J. Edwards (Ed.), *Proceedings of the Twenty-Third Australasian Computer Science Conference* (pp. 88–94). Los Alamitos, CA: IEEE Computer Society.
- Lamport, L. (1994). *LaTeX: A document preparation system*. Reading, MA: Addison-Wesley.
- Macnamara, J. (1996). *The modern presenter's handbook*. Sydney: Prentice-Hall.
- McGregor, H., Saunders, S., Fry, K., & Taylor, E. (2000). Designing a system for the development of communication abilities within an engineering context. *Australian Journal of Communication*, 27, 83–94.
- Mohan, T., McGregor, H., & Strano, Z. (1997). *Communicating! — Theory and practice*. New York: Harcourt Brace.
- Peyton Jones, S.L., Hughes, J., & Launchbury, J. (1993). How to give a good research talk. *SIGPLAN Notices*, 28, 9 — 12.
- Steffe, L.P., & Gale, J. (Eds.). (1995). *Constructivism in education*. Hillsdale, NJ: Lawrence Erlbaum.
- van Leunen, M.C. (1992). *A handbook for scholars* (2nd ed.). New York: Oxford University Press.
- Winn, W.D. (1991). The assumptions of constructivism and instructional design. *Educational Technology*, 31, 38–40.
- Young, A., & Fulwiler, T. (Eds.). (1986). *Writing across the curriculum*. Portsmouth, NH: Boynton/Cook.
- Zobel, J. (1997). *Writing for computer science*. New York: Springer.