The Impact of Campus-wide Portable Computing on Computer Science Education

Report of the ITiCSE'98 Working Group on Campus-wide Portable Computing

Stan J. Thomas (chair)
Wake Forest University, USA
sjt@wfu.edu

Tomohiro Nishida
Osaka University, Japan
t-nisida@ics.es.osaka-u.ac.jp

Cary Laxer
Rose-Hulman Institute of Technology, USA
laxer@cs.rose-hulman.edu

Helen Sherlock
National University of Ireland, Maynooth, Ireland
Helen.Sherlock@may.ie

ABSTRACT

At least fifty colleges and universities throughout the world have initiated plans under which all students buy, lease, or provide their own portable/laptop computer. The impact of these initiatives on the general educational environment has been widely discussed; however, the impact on the delivery of computer science education has not been. In this working group report we discuss a number of issues pertaining to computer science education in light of campus-wide computing initiatives. Our report relates experiences of faculty already involved in such endeavors plus ideas regarding the future of portable computers in computer science education. As such, it will be beneficial to computer science programmes currently involved in campus-wide computing initiatives as well as those that may become involved in the near future.

1. INTRODUCTION

For reasons ranging from economics to changing trends in pedagogy, colleges and universities around the world are adopting computing initiatives that require every student to buy, lease, or otherwise provide their own portable/laptop computer. Dr. Ray Brown of North Dakota State University reports that there are now at least fifty such institutions (see URL in [2] for the current list). Many of these campus-wide initiatives have involved complete revisions to campus networks and instructional facilities [3][4]. Some have required extensive faculty computer training and support programmes [7]. Others have been coupled with broad initiatives to improve classroom instruction by moving away from the lecture format towards increased collaborative learning [8].

The progress of these campus-wide initiatives has been reported widely in publications ranging from newspapers to the *Communications of the ACM* to the Internet [6][5][13]. However, little has been written specifically about their impact on computer science education. What certainly is being discussed is how the Web, Java, and collaborative learning are creating a "paradigm shift" in computer science education [1][11][16]. In this report we focus on how campus-wide laptop computing initiatives contribute to the overarching paradigm shifts currently underway. The discussion is organized around a number of issues that arise when considering the question "how can computer science educators best take advantage of ubiquitous portable computers to accomplish our educational goals?" The specific issues discussed are pedagogy, curriculum, standardization, and facilities.

2. ISSUES

2.1 Pedagogy

There are academic disciplines, particularly in the humanities, in which the availability of portable computers appears to be making a major impact on instruction [17]. Faculty members are excitedly inventing ways to use these tools to enhance instruction. In computer science we are somewhat inured to this excitement because of our long-term, daily exposure to computers. How can computer science faculty be motivated to take advantage of opportunities available or, to quote Tom O'Dwyer, Director-General, DHXXII, Education Training and Youth, European Commission, from the opening keynote at the ITiCSE'98/ACTC'98 conference, "to practice what we preach"?

Campus-wide portable computing would appear to be an important step in harnessing the new technology to improve computer science education. It has been pointed out that the interactive features of computers are an enhancement to learning [13]. But technology by itself will not achieve the required paradigm shift. This has been demonstrated many times over the past fifty years. Historically, computer-assisted instruction has not delivered on its promise to revolutionize education. In business, the anticipated productivity gains from 4GLs, CASE, ICASE, and workflow software have not materialized. In fact, the greatest software productivity increase occurred when programming moved from machine-level to high level language. From these experiences we can learn that technology is merely an enabler to improvement. It cannot ensure improvements by itself [15]. In education, this means that it must be used in partnership with innovative pedagogy [8]. The difficulty lies in making the innovative pedagogy a reality.

It would be nice if one could follow a pre-defined set of steps that ensure successful innovative pedagogy. Unfortunately, they do not exist. What is obvious is that much proactive experimentation will be required. What works for one institution may fail for another. Each has its own unique set of students, teachers, objectives, strengths, and constraints. Nonetheless, there appears to be agreement that the two central roles must change—that of teacher and student. In the new paradigm, teachers become facilitators and students become teachers [12]. A class becomes a much more interactive environment in which the students work at the computer, exploring solutions to problems. The teacher poses the problem, guides and encourages, brings solutions found by students to the attention of the rest of the class. The students sometimes work singly, but more often in groups of two or more. It is a significant departure from the traditional lecture format. We shall now discuss both the roles and the implications of the changes.

2.1.1 Teachers

For teachers, the use of any new technology should support one central objective, that of becoming better teachers. The availability of ubiquitous computing should particularly enhance the teaching of computer science. As stated above, to fully exploit the potential of campus-wide portable computing, the role changes from teacher to facilitator, or from the “sage on the stage” to the “guide on the side”. The new role requires a corresponding new set of skills. These include the creation of a learning environment, management of collaborative efforts, familiarity with the new technology, greater interaction and sharing with other educators in the field, and flexibility in course design and execution.

The sage on the stage has all the answers. Each lecture is carefully planned and timed to cover a particular subject. The students work from recommended textbooks. The delivery is one-to-many with little feedback or interruption. In contrast, the guide on the side allows the students access to vast banks of information, the most powerful being the World Wide Web. The class time may still partially work on a one-to-many delivery, but the class now has the power to learn much more than the lecture material. The teacher has to deal with no longer being the “expert” [12]. In essence, the teacher has to create that learning situation, where students are empowered and encouraged to learn and communicate their newfound knowledge. The teacher will not have all the answers. In acknowledging this and actively encouraging the class to find the answers elsewhere, the teacher is clearly signaling to the class that it is acceptable to admit ignorance. The teacher is creating a better learning environment. But this is an appreciably difficult step for the former expert.

The merits of collaborative or team-based work have long been recognised but little used in computer science education. It has been shown that people are more productive working in pairs than singly [14]. In-class collaboration and out-of-class team projects are facilitated by the new technology. However, faculty have little experience managing collaborative efforts. It has been discovered, through experience, that in-class collaboration takes longer than anticipated and requires careful planning [15]. Additional issues arising include the danger of plagiarism [9] and the difficulty of assessing team-based projects [10][12]. Plagiarism has always been an issue, even in closed laboratories, where constant supervision is often impossible. However, the source for plagiarized materials is extended now to include not just fellow students but anyone in the world who wishes to post their work on the Web. In such a scenario, it is extremely difficult for a teacher who suspects plagiarism to locate the source of the material. Assessment of team-based projects is also a difficult area to judge. Should the teacher mark a project assuming peer pressure has forced all team members to contribute more or less equally? Or should each team member be evaluated separately on the basis of his or her perceived workload and contribution? Both have advantages, the former being more straightforward, the latter possibly being more equitable.

Even among computer science educators, there is a need for extra training to fully utilize the availability of computers to all students. Some institutions have approached this by retraining faculty before campus-wide computing is introduced [8]. The success of the initiative depends to a large extent on all faculty enthusiastically adopting the new approach to instruction. Resistance to change by some faculty will be encountered but can be reduced if the training is comprehensive. New software will be essential to fully exploit ubiquitous computing. For example, software can enable the facilitator to view progress in the electronic classrooms, make on-line assessment and corrections, and monitor on-line discussion groups for out-of-class team projects [13]. Faculty must be competent in the technology to enable them to present the right level of challenge to their students. Time is needed to ensure this competence and to revise course content to take full advantage of the technology. A concern among many busy teachers is finding time to learn new technology and to develop tools. Will this time emerge from a reduced teaching load or be taken from their allocated professional development time? Or will it simply be seen as an extra task to be completed while still fulfilling existing obligations? If the latter, then it is unlikely that the full benefits of the technology will be exploited. Traditional lecture formats and existing course content will remain and the laptop will simply be an electronic pen and paper.

As well as encouraging students to take advantage of the vast resources now available to them, faculty must also rethink their method of working. Traditionally each faculty member created their own set of lecture notes working from one or more textbooks. Now that they are no longer bound by geographical constraints, sharing materials across institutions, should become the norm. Examples of course materials currently available on the web include algorithm animation and simulations; but there is potential for much more. Teachers can learn from the experience of others who have been pioneers in campus-wide computing.

Much experimentation will be necessary before we as educators will know if we have achieved a paradigm shift. What is certain is that course design and execution will require flexibility. The basic curriculum may remain unchanged, but the accessibility of more and more information allows depth of knowledge to be acquired in new areas. Recognising and rewarding students who display initiative then becomes even more important than at present. If students are encouraged to use technology to find out new information, can examinations be structured in such a way to reward them for their extracurricular knowledge? A rigid examination structure as it exists in some institutions does not lend itself easily to this. As educators, we would also expect that the standards achieved by the students would be raised by their constant access to the technology. This may be reflected in more demanding projects, particularly final year projects, and a higher standard in all examinations.

2.1.2 Students

The role of the student can change dramatically with the advent of campus-wide computing. If faculty adopt the electronic classroom and facilitator role, then the student must assume more responsibility for learning. As stated previously, faculty will have higher expectations of their students in this new environment. It is important that the students are fully apprised of this additional responsibility before they enroll [10]. To complement the extra responsibility, students will now have constant access to a great deal of additional information. Simple actions such as posting class notes on the network are immediately more effective due to the universal access. For students to fully succeed, the laptop must become an active learning tool, a vital element of their education. There are indications that people learn best by constructing their own knowledge [16]. The laptop can aid the student in this active knowledge construction, in contrast to traditional lectures where the student is passively led and constrained by the lecture content.

The change from student to “self-teacher” will take time and careful nurturing by faculty to accomplish successfully. The learning environment we already discussed must exist to reward effort and encourage exploration. It has been shown that under some conditions students ask more questions via e-mail than in class [8]. A well-maintained bulletin board, list service, or newsgroup will be a benefit to all students, as well as providing the teacher with a good understanding of student comprehension. Others have successfully experimented with on-line anonymous question and answer sessions, facilitating those who are still too shy to acknowledge lack of understanding and encouraging all the class to voice their ideas on a topic [14].

Ubiquitous computing also reduces the time spent by students in college laboratories. It is much more likely that students will do assignments in their own residence on their own time. This raises a number of issues. In laboratories, students may come into contact with faculty, teaching assistants, or other students, and have an opportunity to discuss aspects of the course. On-line communication facilities, particularly e-mail and on-line discussion groups, are an invaluable source of communication but can never replace face-to-face contact. In verbal communication, topics arise naturally that would normally not be discussed on-line. These dialogues provide useful information to both students and teachers. There is a risk that students will become so dependent on their computers that they lose the ability to communicate orally. This may be handled by special communication and presentation courses, or by teachers occasionally insisting on ‘laptop-free’ sessions. Working from the privacy of their own residence may result in students taking more active involvement in aspects of the web, e.g. posting their opinions to news-groups.

As with faculty, there is a learning curve for students to familiarize themselves with the new technology. It must be decided if this learning curve should be handled by a special introductory course in the first semester, or as part of the normal curriculum. Faculty have an important role to play in encouraging students to explore the potential of the technology. If faculty lead in the right direction, students can successfully follow and computer science education can be enhanced by the ubiquitous availability of the technology.

2.2 Curriculum

Many of the schools that have adopted campus-wide portable computer initiatives are liberal-arts institutions that have traditionally attracted students with similar interests. Computer science curricula in such schools often reflect this environment—they emphasize general concepts, theory, and abstraction. What, if any, will be the impact of a student population that is more computer literate than in the past and who live with their computers twenty-four hours a day? Will these students demand that computer science curricula become more pragmatic and less theoretical? And what is the appropriate response?

Several engineering schools have also adopted campus-wide portable computer initiatives. Like their liberal-arts counterparts, these schools also attract students with common interests. However, these interests tend to be more applied than theoretical. Frequently, these schools have a computer science service course component, teaching introductory computer science to all majors at the schools. Can the service courses be made more interesting, interactive, and worthwhile for all students when they have their own computers?

No matter what type of school, students will demand that their laptop computers be used in their courses if they are required to buy them. To respond to this demand, faculty will have to rework courses to incorporate the use of laptops. For some courses, this may be a trivial exercise. For other courses, extensive work may be involved. Faculty will need sufficient lead-time to plan for the effective use of the laptops in their courses.

We anticipate that students will become comfortable using their own computers and will therefore spend more time on their portables than they will on laboratory-based machines. This might encourage students to explore their computer's capabilities and the problems they are asked to solve beyond the minimum requirements set forth by the instructor. How should this independently motivated work be rewarded? Should extra credit be granted on a project's grade? Should a course grade be adjusted to reflect the student's extra efforts?

2.3 Configuration Standardization Issues

Anecdotal evidence from schools with campus-wide portable computing initiatives leads us to believe that many of the benefits of such programmes—for the campus as a whole—derive from the standardization of the hardware configuration and the initial software load. Computer science educators may not reach the same conclusion. The common computing platform will probably be selected by committee, possibly with no computer science representation, to meet the common needs of the student body. The committee's decision may be based primarily on acquiring the newest, fastest hardware for the money available. There is no assurance that the specific needs of computer science will be addressed or met. In fact, computer science educators may well question the adequacy of any single platform.

Computer Science majors should be given a broad view of the world of computing. The technology workplace requires people to perform daily in complex networks where they must use UNIX, Windows, DOS, Macintosh, and other user interfaces. The successful computer science student must not only be able to use multiple operating systems, but needs to understand underlying differences in their design and implementation. To address these diverse educational needs, departments may find it necessary to continue to provide laboratories that include operating systems other than the campus standard. An alternative approach is to ensure that the standard platform has sufficient memory and free disk space so that students can experiment with alternative operating systems while still adhering to the campus "standard."

Whatever the intended use, the life cycle of one version of hardware or software is brief. Students from different academic years may, in fact, have different platforms. Thus, whenever a class contains students from different academic years and/or transfer students, problems are likely to arise. It is not practical to upgrade hardware every year, but software can, and should, be upgraded regularly. In courses requiring programming, for example, experience has shown that the instructor's job is greatly simplified when all students use not only the same compiler but also the same version.

However, frequent software upgrades are not free. Decisions must be made as to whether students pay the upgrade costs directly to the vendors, pay the costs through lab fees, or bear the costs indirectly through increased tuition. Many schools with laptop programmes are finding that the total cost of software upgrades is reduced by negotiating quantity or site licenses with software vendors.

Yet another major issue is that of purchasing and upgrading the standard platform for all faculty. Ideally, the instructor's platform should match that of all the students in the class. However, that is not practical since many faculty teach courses at different levels each term. While it is cost prohibitive to upgrade all faculty hardware each year, annual software upgrades for the entire campus, including the faculty, will greatly reduce the impact of having multiple platforms in the same class.

2.4 Facilities

2.4.1 Networks

Omnipresent portable computers require convenient, reliable, high speed network access from classrooms, residences, study areas, libraries, even from hallways—any place a person may be for more than a few minutes. Computer science faculty and students frequently demand greater bandwidth and network capacity than those in other disciplines do. In some institutions, the cost of expanding network access within departmental facilities may have to be borne by the computer science department.

Some institutions with ongoing campus-wide portable computing initiatives are experimenting with wireless, mobile network technologies. Such wireless networks may alleviate the inconvenience of extensive physical wiring but, even so, networking costs are not likely to decrease in the near term.

2.4.2 Classrooms

At the very minimum, the use of computers during in-class lectures necessitates having network access and projection equipment for the teacher to use for instruction and demonstrations. However, a change in pedagogical style from passive learning to active learning enhanced by technology calls for an altogether different learning environment from the traditional lecture room. The University of Maryland has been experimenting with "electronic classrooms" for over seven years [14]. They recommend four basic physical design decisions for using non-portable computers in the classroom as active learning tools:

- Two students per computer to encourage discussion.
- Monitors partially recessed into the desks for good sight lines.
- Computers located in an adjoining room to reduce bulk, heat, and noise, while improving security.
- Network connectivity to reduce the need for floppy disk access and printing in the classroom.

Although the Maryland guidelines assume classrooms are permanently equipped with computers, they could be readily adapted to portable computing.

A second project, also aimed at improving the teaching/learning environment through the use of technology, has been underway at Rensselaer Polytechnic Institute since 1993 [18]. The Rensselaer group has reported very positive results using its "studio model" of instruction for teaching physics and mathematics. The studio model involves not only an electronic classroom layout but also multimedia course material and custom designed software called the CUPLE system. Although the studio model approach may also be suitable for teaching computer science, no such efforts have been reported in the literature. Although initially expensive, the claim is that the method can reduce contact hours while improving learning.

The use of portable computers in the traditional lecture classroom gives rise to several potential problems. It may make it more difficult than ever for the instructor to capture and retain the interest and participation of all students, especially when network access is permitted and the instructor sees only the back of the computers. Some lecturers, having experienced this problem, have considered asking for mirrors to be installed on the rear walls of classrooms. An easily overlooked problem with computers in the classroom is the effect of numerous keyboards clicking simultaneously. These human factor issues are important and many are specifically addressed in the implementation of the studio model classroom described above.

2.4.3 Labs

Computer science programmes have traditionally spent a substantial portion of their budgets purchasing and maintaining laboratories equipped with workstations or personal computers. When every student has a portable personal computer, are laboratory facilities still necessary? If so, should the labs be equipped in the traditional way, or is it adequate to provide network and electrical connections and have the students provide the computers?

Budget-conscious administrators may believe that ubiquitous portable computers supplant the need for laboratory facilities. The experience of schools with portable computing initiatives is that fewer general-purpose computer labs are necessary than before; however, there remains a need for laboratories with specialized computing equipment to support courses such as computer graphics and parallel computing. In addition, we believe that laboratory machines must still be provided for computer science students in order to expose them to a different environment and operating system than that available on their personal computers. Currently, the most common scenario is to provide workstation laboratories using some version of UNIX while the students run a Microsoft, or Macintosh, operating system on their portable personal computers. Certainly, a dual-boot configuration is an alternative if disk space is adequate and the university's "standardization" guidelines can still be met.

2.4.4 Support

No matter what hardware and classroom facilities are involved, issues of maintenance and support must not be overlooked. It is frustrating, yet tolerable, to arrive at the classroom and have the overhead projector bulb fail or to be out of chalk. It is quite another thing to prepare an exercise for your students to do during class on their computers and have the network or computer projection equipment fail. In addition to campus-level maintenance for large numbers of computers, knowledgeable, responsive support personnel must be available at the departmental or building level to respond to unexpected problems.

Fortunately, it appears that campus-wide laptop programmes can reduce the pressure on departments to provide and maintain labs on a twenty-four hour basis. When every student has their own laptop and easy access to printing, contention for computer labs near project deadline dates can be reduced or eliminated altogether.

2.5 Other Issues

For any campus-wide laptop plan to succeed there must be a well-designed training programme to introduce students to the capabilities of their new computer. This training will typically include introductory instruction in the standard suite of software tools on the machine, a discussion of appropriate and ethical use of computers and the Internet, plus basic care and operation principles (for example, how to prolong battery life).

The need for such training may impact the computer science program in various ways, depending on how and when the training is conducted. For example, is the training conducted during first year orientation or is the computer science faculty expected to add this additional material to a required first year computing course? If the time is taken from first year orientation, who does the training? Are there even enough computer science faculty to conduct the training? Are computer science faculty required to participate or are they permitted to volunteer?

3. CONCLUSION

One impact of campus-wide portable computing on computer science education is that it tends to change the communication patterns between students and faculty. The amount of electronic communication is likely to increase, but the verbal interaction outside of class may decrease.

The availability of ubiquitous portable computing, like any technology, cannot itself engender a paradigm shift in computer science education. However there remains the potential that such initiatives, when employed creatively, will contribute significantly to increased collaborative and active learning.

4. ACKNOWLEDGMENTS

We would like to acknowledge the contributions of Dr. Jennifer Burg of Wake Forest University and Dr. Carolyn Watters of Acadia University who participated in workgroup discussions but were unable to attend the conference.

5. REFERENCES

[1] Boroni, M.C., Goosey, F.W., Grinder, M.T., and Ross, R.J. A Paradigm Shift! The Internet, the Web, Browsers, Java, and the Future of Computer Science Education, *Proceedings of SIGCSE '98* (Atlanta GA, April 1998), ACM Press, 145-152.

[2] Brown, R., Notebook Colleges & Universities, Accessed 15 July 1998. Available at http://www.vcsu. nodak.edu/offices/itc/notebooks/other.htm

[3] Brown, D.G. A Plan for the Class of 2000. *Multiversity* (Spring 1997), 12-16.

[4] Brown, D.G., Burg, J.J., and Dominick, J.L., A Strategic Plan for Ubiquitous Laptop Computing, *Comm. of the ACM* 41, 1, January 1998, 26-31.

[5] Burg, J.J., and Thomas, S.J. Computers Across Campus, *Communications of the ACM* 41, 1, January 1998, (special section).

[6] Burritt, C. Wired 101, *Atlanta Journal*, May 11, 1997.

[7] Candiotti, A., and Clarke, N. Combining Universal Access with Faculty Development and Academic Facilities, *Communications of the ACM* 41, 1, January 1998, 36-40.

[8] Deden, A. Computers and Systemic Change in Higher Education, *Communications of the ACM* 41, 1, January 1998, 58-63.

[9] Joy, M., *et al*. Software Standards in Undergraduate Computing Courses, *Journal of Computer Assisted Learning* 12, 1996, 103-113.

[10] Lawhead, P.B., *et al*., The Web and distance learning: what is appropriate and what is not, Report of the ITiCSE '97 Working Group on the Web and Distance Learning, in *Working Group Reports and Supplemental Proceedings of SIGCSE/SIGCUE ITiCSE'97*, ACM Press, 27-37

[11] Naps, T., *et al*. Using the WWW as the Delivery Mechanism for Interactive, Visualization-based Instructional Modules, Report of the ITiCSE '97 Working Group on Visualization, in *Working Group Reports and Supplemental Proceedings of SIGCSE/SIGCUE ITiCSE'97*, ACM Press, 13-26.

[12] Paxton, J.T. Webucation: Using The Web As A Classroom Tool, *Proceedings of SIGCSE '96* (Philadelphia PA, February 1996), ACM Press, 295-289

[13] Pinheiro, E. Introducing Mobile Computing to the College Campus. Accessed 15 July 1998. Available http://www.hied.ibm.com/segments/ tpu.html

[14] Shneiderman, B., Alavi, M., Norman, K., and Borkowski, E.Y. Windows of Opportunity in Electronic Classrooms, *Communications of the ACM* 38, 11, November 1995, 19-24.

[15] Shneiderman, B. Educational Aspirins for Web Fever: Human Values for Shaping Educational Technologies in SIGCSE '98 Symposium Luncheon Presentation, February 27, 1998

[16] Sollohub, C. Computer Technology in the Classroom: New Paradigms! New Designs! Better Software? in *Proceedings of the 4th Annual Midwestern Conference of the Consortium for Computing in Small Colleges*, 13, 2, November 1997, 229-236.

[17] Watters, C., Conley, M., and Alexander, C. The Digital Agora: Using Technology for Learning in the Social Sciences, *Communications of the ACM* 41, 1, January 1998, 50-57.

[18] Wilson, J.M. The CUPLE Physics Studio. Accessed 19 August 1998. Available at http://www.educom.edu/ program/nlii/articles/wilson.html