Assessment of the Student’s Design & Problem Solving Experience
Utilizing Notebook Computers and a Wireless Environment

Stuart Steele
Polytechnic University
Brooklyn, NY

Abstract

The purpose of this paper is to discuss how students utilizing notebook computers and a wireless environment can improve their design experience in engineering and technology based courses including computer science and project courses. Issues affecting both notebook use and wireless operation will be presented. Teaching models will be explored in order to emphasize the problem solving and design experience. Both lecture and laboratory settings will be addressed. The impact of mobile computing with notebooks is reviewed from the viewpoint of team efforts and remote activity. Results from two courses in the Fall of 2000-2001 school year at Polytechnic University will be shown, where notebook computers are now required and many classrooms, labs and campus areas utilize wireless operation. Covered are freshman and upper level courses.

I. Introduction

Recently there has been a great interest and desire to make notebook (laptop) computers available for students at many universities. This is to provide a mobile type computing approach for better communication between faculty and staff and to improve the learning experience in the classroom. The advent of wireless technology and the recent 11 MHZ standard has allowed wireless technology to be added to the notebook computer for classroom and outside work. This is in addition to the present computer laboratories existing currently.

Polytechnic University has been upgrading its wired environment over the last few years and in 1999 the school was ranked number 75 in the "100 most wired" schools due to the extensive amount of new computing capability. In addition to a high speed wired environment the advent of mobile computing using notebook computers and wireless work areas and classrooms is changing the culture at the University. Effective September 2000 all entering freshman are required to obtain a notebook computer. This computer is supplied by the university and the student pays fees on a semester basis for its use, eventually owning the machine. This year the computer was an IBM ThinkPad T-series computer. This machine has a 700 Mhz clock, 128Mb of memory, a 12GB hard drive and a CD/DVD in addition to normal interfaces and a wireless card. The software is quite extensive and includes Labview, Common Space, MatLab, AutoCAD Lite and a full Microsoft Suite including development tools. In addition to these changes the total undergraduate program (all four years) at Polytechnic University has been revised as
Curriculum 2000 with each semester being a set of four courses, four credits each, regardless of major. The notebook computer is in support of all freshman courses. This includes Introduction to Engineering, Calculus I and II, two Computer Science courses, Physics, Chemistry and various Humanities/Social Science courses. One course of interest is CS 1114 an *Introduction to Programming and Problem Solving* taken by all entering students. This is a problem solving type course in addition to writing working computer programs. Designing a working program is of major importance here. This class with six lectures and 15 labs covered close to 500 students. Instructor styles differ for the lectures, while the computer laboratories utilize the same format and are PC based in addition to notebook computers. This facilitates a number of students who were taking the course without notebook computers.

In addition to this first level course the use of notebook computers in the curriculum encouraged many upper-level students who had notebook computers of their own but needed a wireless card to purchase one and do their projects utilizing that technology. CS 398, the computer science senior project course, changed as many students used their notebook computers with wireless technology to solve project issues. As many of these students are working and commute to school this flexibility was important since team development is stressed in this class.

This paper addresses notebook and wireless operation experience with both CS 1114 *Programming and Problem Solving* and CS 398 *Advanced Senior Project*.

II. Environment

For effective use of notebook computers, a strong support environment is required for the students. A separate notebook computer help desk is in place to answer questions on why the notebook is not working or general questions on notebook use and procedures. The help desk personnel are capable of re-imaging the software on the notebook computer if required.

For students working at home a portion of the Polytechnic University network operates as an ISP (Internet Service Supplier). Dial up networking configurations are already established on all notebook computers. Once a connection has been established, students can then access the university’s Intranet, Internet, e-mail, on-line courses, etc. ¹

III. Background and Overview of CS 1114 Classes (*Introduction to Programming and Problem Solving*)

The new 4 credit CS 1114 *Introduction to Programming and Problem Solving* and CS 1124 *Object Oriented Programming* courses replaced the previous three credit freshman courses. They are part of a restructuring of the entire freshman program. This has provided a chance to increase the amount of hands-on programming and to increase the amount of object oriented programming concepts before the more theoretical third course in the sequence, CS 2134, *Data Structures and Algorithms*. ²
All students at Polytechnic take the first course, while Electrical Engineering students take two courses and all three courses are taken by Computer Science and Computer Engineering students. In the first course the students are segregated in experienced and inexperienced sections.  

As a goal, students are able to design and create, basic programs written in C++. General topics include the fundamentals of programming, good software development practices and problem solving skills.

The design involved for most programming assignments is quite straightforward in this first course. The twenty homework assignments required students to read and understand increasingly more complicated problem descriptions, but each was relatively short and self contained. It is believed that because of the diversity of our student body, a great deal of practice converting technical English problem descriptions into working code is essential.

Introductory programming is a language learning activity involving essentially the vocabulary and grammar or syntax of a new foreign language, in this case C++. Implementation of a problem solution requires proper use of these new language statements. Beginners will not be able to converse with the computer using this language unless they do so flawlessly. Since that is unlikely to happen, a major portion of the learning experience involves debugging of the student’s code.

Students are therefore constantly involved in problem solving for every statement of their code which generates a syntax error. This is not a trivial skill to acquire. In the past, online tests have been given in a class of this type. They are open book, open notes and full use of the associated online help files. However, students are allowed to ask questions during the exam, provided that the questions can be answered with a "yes" or "no". Often, very few students can play a game of "20 questions" effectively in order to narrow down the possible causes for the syntax error. Since this is one of the main techniques any software developer must apply in debugging his/her own code, it is a skill that needs to be practiced and learned as early as possible.

IV. Results of CS 1114 Class

Due to the increased hands-on activities provided by the new curriculum design, students are much more involved.

The following applies to one of the teaching models used in this class (class type 2). Students are required to attend the lecture with their notebooks, and to "take notes" on those machines. Lectures were presented entirely in the Visual C++ environment and consisted of development of working code supplemented with appropriate comments (for what would otherwise be traditional notes). Although many students did not have the keyboarding skills to copy verbatim, they all attempted to do so. As with other lecture courses, students need to develop note taking skills that include the ability to write down the essential material while skipping details.
Many editor features such as "cut and paste" make note taking more efficient once students learn those features.

Live code creation has several nice features. Students can see a problem solution develop without being able to glance ahead at the finished product. If the professor includes "unintentional" or "intentional" design errors during a lecture, the students also get to see that they are not expected to "get it right" on the first try. They also see that there are a multitude of possible problem solutions that can be and should be explored. Students are encouraged to "help" by spotting typos or other mistakes before the professor can fix them.

As students try to copy the professor's code from the screen, they inevitably make mistakes. This is considered a good thing. They get more practice detecting syntax errors, and fixing them quickly by looking at the correct answers on the screen. However, if they get an error that is too difficult to fix quickly, they have the option of asking a question. Since it is unlikely that they will be the only student to ever make that error, this is a good chance to proactively warn the other students.

Experimenting with a piece of language syntax is essential to problem solving and design. In some ways, this style of lecturing is more efficient in the amount of material which can be presented. It is also more efficient since any material produced in class can be saved and distributed to the class rather than erased from the blackboard by the next instructor in the room. However, the object is not to move on too fast for the entire class so in practice, less material is presented, but the opportunity for learning is optimized.

Students attend a three hour lab each week. This lab is subdivided into three parts. The first is a one hour programming and problem solving exercise. Students are required to write code to solve several short language related problems reinforcing concepts discussed in the lectures. For those students whose Instructors did not use the notebooks in class in the way described this would be the first chance they had at an instructor supported class where they write code. The others would see this as a more individualized environment, since each lab was much smaller than the lecture class.

The second hour of the lab required the students to write a program correctly solving five small problems in 30 minutes. The short amount of time is intentional in order to force students to develop problem solving skills. They might quickly try an approach, test it, and reject it if it does not work. They can use online help or reuse their old code. They need to decide which is more useful for a given problem. There was an obvious and substantial improvement in the scores for these exams, over the course of the semester.

The third hour of the lab is reserved for individual help on the homework problems. The students who used the notebooks in class felt more confident, but student grade results did not indicate a preference. Polytechnic University also has a major campus on Long Island at Farmingdale. In this group of 100 students, the CS freshman laboratories were implemented using laptops only (class type 1).

*Proceedings of the 2001 American Society for Engineering Education Annual Conference & Exposition*
*Copyright © 2001, American Society for Engineering Education*
General feedback from the student notebook computer user group indicated a wide variance in class use of the notebook computer but heavy use out of class, among groups and individuals. Most freshman take four classes that meet twice a week (3 hours) and there were additional labs and recitations (8-10 classes per week on average). Fifty percent of the user group used the computer in class up to three hours with a wide range of utilization outside of class. This is essentially the utilization of an effective wireless environment within class rooms and library buildings and outside areas.

For CS 1114, Table I shows the four class types or teaching models, used at Polytechnic University in the Fall of 2000 and the notebook use. Use of Notebooks in large Lectures indicate that the instructor and the students are both designing and solving problems using notebooks at the same time. Notebooks as Labs indicate the full laboratory utilization is laptops. Notebooks in Labs mean that the basic PC laboratory is complemented by notebooks. Notebooks as tools indicate that the students are using the notebook extensively out of class in a very mobile manner. In all cases it is possible for the instructor to use a power-point generated presentation for the class that is then web accessible.

Students felt that the mobile aspects of the notebook computers allowed them to be more efficient in working together and problem solving, especially if not in the facilities of the University. The most negative issue with regards to notebook utilization is that many students use the notebook for other things during class and are not disciplined to follow the instructor when required, becoming easily distracted.

Table I - Notebook Usage and Class Type (Teaching Models) - CS 1114

<table>
<thead>
<tr>
<th>Notebook Use</th>
<th>Class Type 1</th>
<th>Class Type 2</th>
<th>Class Type 3</th>
<th>Class Type 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notebooks in Large Lectures</td>
<td>NO *</td>
<td>YES</td>
<td>NO *</td>
<td>NO *</td>
</tr>
<tr>
<td>Notebooks as Labs</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>Notebooks in Labs (PC Labs)</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>Notebooks as an efficient tool</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>

No relationship was identified between the final grade distribution and the class type.

The Fall 2000 Student surveys for course assessment showed that the Class Type I configuration scored 20% higher than the other approaches. The other type’s scores were about even. Many variables, such as instructor and general class material may impact this. In Class Type 2, of the students commenting on notebook use, 60% had favorable comments.
V. Background and Overview of CS 398 (Advanced Senior Project)

The senior project represents the capstone design course of the Computer and Information Science (CIS) curriculum. The senior project provides an opportunity for students to apply the knowledge and skills gained in other courses as they synthesize a solution to a significant, realistic problem, plan and manage a software development project and implement the solution using state of the art methodologies, techniques and tools. Senior projects are carried out on a team basis (individual projects are only permitted with permission of the instructor). After completing the project work, the student prepares a demonstration of the application to the instructor and/or simulated clients for evaluation and feedback. The applications developed typically represent solutions to real world problems that are either assigned by the instructor or proposed by the project team. Projects are evaluated for quality and completeness: project planning and management (documentation and execution), requirements, design and architecture specification, conformance with standards and implementation completeness.

Project demonstrations are conducted in the CIS Software Engineering or Distributed and Parallel Processing Laboratories. These laboratories are comprised of state-of-the-art 3 tier technologies (hardware, networking, and application development tools) with required software (middleware, languages, database management systems, capacity and performance tools, and a full complement of automated life cycle development tools). Currently the laboratory relies on fixed base resources (application and database servers, clients and networking).

Students are encouraged to gain knowledge about how they solve problems through "best practices", in which they not only perform the work, but analyze how it was accomplished. To achieve the goal of the senior project to operate within time, cost and resource constraints the following criteria is used in selecting a project:

- The problem is within the context of a larger development effort. This requires establishment and tracking of project scope and interfacing with an external environment.
- Clients outside the development team are the intended users of the software systems
- Students have the opportunity to demonstrate individual achievement. Each team member has a unique set of documented roles throughout the project life cycle. These roles are selected and the project organization is developed by the project team and specified in the project plan.
- The project presents a significant state-of-the-art technical challenge
- The students have the opportunity to apply knowledge and skills gained in previous courses.
- The content of the system is implemented in the laboratory environment.
For Spring and Fall of 2000 the combined Software Engineering Laboratory and the Parallel and Distributed Computer Laboratory environment had a full architectural component and artifacts to support a wide range of two and three tier client server systems. This included a Database server with UNIX, Oracle, SQL, and Sun servers; a Network both wired and wireless with TCP/IP, ATM and the Internet; An Application Server with Unix, Windows NT, C++, Java Beans and Enterprise Java Beans; Middleware, with UNIX tools, Java tools, and TCP/IP multicast; Clients with PCs and Laptops; Application software with Cold Fusion, Apache, SQL, mySQL, Access, Java Beans, Applets, Servlets; and a Development environment with Rational Suites, visual tools, C++ and Java.

A typical list of projects include commercial applications such as security authentication, local complex games, stock market simulation, bank simulation, college registration system with artificial intelligence, distribution systems, bond trading and tracking systems, internet routing optimization, internet search algorithms, wireless administration and a distributed text classification and comparison system.

VI. Results of CS 398

An interesting observation with this experience is that the trend of technology usage (Table 2) highlights the system changes to wireless and notebooks. This Fall many students, having their own notebooks, found that by buying the wireless card and downloading software they could use the notebook computer for project planning and technical design of their team project. The team impact of this was exceptional since the student group could work together all having computers and make great progress without being in the Polytechnic Laboratory. Of course the Labs are needed for different levels of integration. This is a design and implementation team teaching model, with instructor reviews along the project time line.

Table 2. Percent (%) Of Projects and New Technology Usage (by year)

<table>
<thead>
<tr>
<th>Technology</th>
<th>1999</th>
<th>2000</th>
<th>2001</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wired Network</td>
<td>90</td>
<td>70</td>
<td>40</td>
</tr>
<tr>
<td>Wireless Network</td>
<td>0</td>
<td>30</td>
<td>60</td>
</tr>
<tr>
<td>PC Clients</td>
<td>98</td>
<td>70</td>
<td>60</td>
</tr>
<tr>
<td>Laptop Clients</td>
<td>2</td>
<td>30</td>
<td>75</td>
</tr>
<tr>
<td>Internet</td>
<td>15</td>
<td>70</td>
<td>90</td>
</tr>
</tbody>
</table>

(*) Note: It is expected that students will be using the Rational Development Environment in a wireless laptop configuration and demonstrate their projects in an integrated PC/ laptop (wired/wireless) configuration.
VIII. Summary

The initial reaction to the requirement and use of notebook computers as part of the curriculum at Polytechnic University has been very positive. In addition the wireless technology is quite effective to support mobile operation. Notebooks are used to support classes, from in class operation to laboratory and home use. The notebook has shown to be a powerful tool to support the learning process. New teaching models are being explored to make more effective use of the notebooks for learning. The overall process is incremental as a new freshman class will enter the University in 2001. In addition, more upperclassmen are also utilizing the notebooks for support of design projects and general class support. The design approach and problem solving ability required for students, is enhanced as they perform more practice and meaningful work using the mobile capability of the notebook computer and wireless environment. Feedback from both the CS 1114 Programming and Problem Solving and the CS 398 Advance Senior Project course support this view. Additional student feedback from the formal course evaluation system of Polytechnic University addresses the positive nature of the notebook computers.

Acknowledgments

The author would like to thank Professors Alan Davis, Ming Leung, Evan Gallagher, Peter Swanson, Ronald Fedderson and Linda Grieco for handling the new CS 1114 course so well. Professor Fred Strauss has taken the lead in enhancing the senior design course experience to a first class level. In addition, a thousand thanks go to the Teaching Assistants working the Labs.

Bibliography
1. URL: http://insight.poly.edu/ Computing Environment at Polytechnic University.

STUART STEELE
Stuart Steele is Industry Professor of Computer Science and Chairman of the Department of Computer and Information Science at Polytechnic University in Brooklyn, NY. He has been active in designing new CS laboratories to support the dynamic changes in the Information Technology Age. He has over thirty years experience in industry and education in computers, control, software engineering and real time systems and served as the Chief Technical Officer of a major large scale systems integrator. Dr. Steele received a B.S. Degree in Electrical Engineering from Bucknell University in 1958 and a M.S. and Ph.D. in Electrical Engineering from Pennsylvania State University in 1965.