

Cell Phones as Tools to Teach Computer Science

Michael VanHilst, Ph.D.

Assistant Professor, Florida Atlantic University, Boca Raton, Florida, USA

Martin Griss, Ph.D.

Adjunct Professor, University of California at Santa Cruz, Santa Cruz, California, USA

Abstract

In this paper we propose the use of cell phones and PDAs as teaching tools and target platforms for teaching undergraduate computer science. The advantages are numerous and compelling.

Cell phones and PDAs are less expensive than PCs. Many students may already have one. The devices support load-and-go course material that students can access anytime and anywhere, including short breaks. With network connectivity, students can use them to collaborate in groups. As a target platform for applications, students can show their work to friends and family, and can use the applications they create in their daily lives. Finally, with so much of our software being developed in far-away places, cell phones present unique economic opportunities for local content and micro-market applications, especially in developing countries.

In this paper we discuss the advantages and opportunities in greater detail, discuss development tools appropriate for students and instructors, and discuss some of the courses currently being taught on PDAs. We find many low or no cost tools available for programming with handhelds. We also find many schools that have started to use handhelds in their courses. Never-the-less, handhelds are still novelties that requires special instruction aside from the main material in the curriculum

Keywords

Computer science, handheld, J2ME, Palm, education

1. Introduction

Today's computer science courses are largely taught using desktop PC's. Students do their assignments on a PC at home, on their personal laptops, or in the school's computer lab. Schools maintain PC labs for their student body, and must frequently update them to stay current with changes in technology and standards.

Handhelds offer a viable and attractive alternative. Many students already carry programmable cell phones, while reasonably capable PDA's are available for less than \$100 (US). By putting school work on these small devices, students are not limited to doing their work at home or in the lab. Especially

important for working students, homework can be done during work breaks, riding or waiting for public transportation, and at other odd moments during the day. Students can also work together in groups, as they do in the lab, but on their own territory and at times when the lab is not open.

The convenience factor is only one of three major factors contributing to the appeal of handheld devices in the curriculum. The other two, coolness and employability, are major motivators, as well.

When compared with doing a CS assignment on a PC, doing the assignment on a handheld is cool. Simple assignments of the type found in, for example, a programming or data structures class, are much closer in scope to the size of a handheld application, like a phonebook or todo list, than to Microsoft Word or Netscape Navigator. Having the handheld ready at hand at all times, makes it easier to share ones classroom accomplishments with family and friends, than it is to drag them into your house to the room with the computer to show them what you have done.

Handhelds also present unique opportunities not available on a PC. Students can develop fun and exciting applications with interactions among their machines. The capabilities of most handhelds encourage students to explore human computer interactions beyond the basic keyboard, mouse and window interface of the PC. Students already use these devices as MP3 players, and cameras will soon be standard on cell phones. An increasing number have GPS capability, enhancing the opportunity for location aware software, maps, etc. Most devices have record and playback capabilities afforded by program access to the microphone. With a handheld device these capabilities are available in the field, which is a lot different than recording or photographing with devices attached to the PC that is stuck to a desk in your room.

Employability is a major concern for CS students, and one that lately has contributed to declining enrollments in undergraduate CS programs, at least in the United States. The industry downturn and talk of outsourcing, at least in the US, gives the impression that computer science is not a promising field to pursue. In many areas the major employers that would offer stable employment are branches of companies that, aside from system administration, do their software in far away places. It is common to hear students say there is little demand for software architects and programmers in our part of the world.

The reader might perform this exercise at home or in the office. Go to your PC's "Start:All Programs" menu. Aside from the games, how many applications are there on your system that aren't updated versions of software that has been around for four or more years? Today, the hot action is in web sites and cell phones. Unlike the market for PC software, there is, as yet, no sense that these markets are saturated or dominated by large players. Moreover, handhelds play a central role in tomorrow's vision of ubiquitous computing.

Like web sites, cell phones and PDA's offer significant economic opportunities for local content and other micro-market applications. Because we carry them always with us, and almost everyone has one, there are many opportunities for them to be useful in niche situations – on job sites or other venues, while performing specific acts, or in specific geographic parts of the world. The types of useful applications students think to write for themselves may well be of interest to others in their region of interest. This is especially true for developing countries where local context, language, idioms, and culture can be key differentiators.

2. Tools

There are several ways to develop software for handheld devices. While most of the powerful tools involve cross-development using a desktop or laptop, there are a number of smaller, promising environments for development on the phone or PDA itself. Probably the most common method of programming handhelds involves a special development environment on a separate PC. Non-native,

cross development is particularly useful for cell phones, where the native runtime environment is especially constrained. Java has a freely available cross development environment for embedded systems, while .Net has a free add-on for Visual Studio.

Java for handheld devices is called Java 2 Micro Edition or J2ME. It comes in different forms, called profiles, depending on the resources available on the device. The profile most suited to current cell phones is called Mobile Information Device Profile or MIDP. In order to reduce the size of libraries and executables, the resource intensive libraries and features are omitted – most notably runtime type information or reflection and data structures with iterators. The standard GUI libraries, AWT and Swing, are replaced with smaller footprint libraries more appropriate to a small device.

The standard development tools for J2ME, called the J2ME Wireless Toolkit, are available from Sun's Java website (Sun, 2004). The development tools run on Windows, Solaris, Linux, or the Java Desktop, and can be run in a text mode to edit, compile, etc., or within the Sun ONE Studio IDE – also free, and now available in a mobile edition.

In .Net and Windows, support for handheld devices is called .Net Compact Framework. Two families of the Windows operating system are supported, PocketPC for PDA's and SmartPhone for cell phones. Development for both systems is based on Visual Studio .Net Professional. Microsoft offers a free 60 day trial copy of Visual Studio, and very favorable educational pricing. Both PocketPC and SmartPhone require an additional SDK add-on that can be freely downloaded from the MSDN web site (Microsoft, 2004).

Both the Windows and Java development environments include emulators that display an interactive image of a handheld device on the PC screen and can be used for debugging. Emulators that match the look of specific devices, and additional software, are often available through the free developer programs of specific cell phone and cellular service vendors. As an added plus, many companies and conferences sponsor student competitions to encourage student development of cell phone or PDA applications.

Metrowerks offers a third option for developing ASM, C, C++, and Embedded C++ applications on a wide variety of platforms, including Palm and Embedded Linux. Metrowerks offers very favorable educational pricing on many versions of its CodeWarrior tools, and versions of CodeWarrior are available on the CD's that come with some PDA programming books (Mykland, 2004). An advantage of Metrowerks is that their tools target many of the microprocessors used in industry and by hobbyists, and not just the systems in popular cell phones and PDA's. A large variety of Palm development tools, including emulators and simulators are also available from Palm (Palm, 2004).

An alternative to PC-based cross development is developing applications natively on the device itself. The Palm OS, for both 68K and ARM platforms, has several open source compilers and interpreters available. The available languages include Pascal (Guillot, 2004), C and ASM (Lawrence, 2004), Scheme (Bayer, 2004), and a reduced version of Python (Collins, 2004). These tools are sufficient to support exercises in common courses such as microprocessors, introductory programming, and data structures. Native tools have the advantage that they allow students to complete their course exercises while on the go. Many devices running the Palm OS are especially inexpensive, costing less than \$99, US.

The PocketPC also supports an impressive set of language tools natively (Keuchel, 2004). ARM assembly, Prolog, Lisp, Perl, and Tcl/TK are available, as is a port of Emacs. Devices running the PocketPC/WinCE operating system are available for less than \$200, US.

For native development environments, however, the Sharp Zaurus is hard to beat. The devices run Linux and sport a slide-out keyboard. The GNU compiler suite runs natively to support assembler, C, and C++.

Java comes installed on the machine and at least two different IDE's are available for native Java development. Precompiled binaries are also available for native development in Perl, Python, Lisp, Scheme, Haskell, and more (Fromme, 2004). Zaurus models are available for as little as \$200, US. Other Linux-based PDA's are available for as little as \$100, US (Softfield, 2004). Linux-based cell phones have recently come on the market, but we are not yet familiar with their capabilities.

Network connectivity may be included in the device, or can be added with a CF card for WiFi, Bluetooth, or simple wired Ethernet. Ethernet CF cards are the least expensive. A network connection can even be established between two devices using PPP over the devices' serial and IrDA ports. Cell phones offer Internet connectivity through the cellular service provider, but this service tends to be expensive, especially for students.

With network connectivity, exercises involving web application development can be supported. SmartPhone, PocketPC, Java, Embedded Linux, and Palm all have available web browsers. There are also small web servers available for all the PDA operating systems – Palm, PocketPC, and Embedded Linux.

A free tool called Plucker (Plkr.org, 2004), compresses normal web pages for a small footprint and easy offline viewing on a handheld, using the Plucker viewer. Plucker is useful for making course notes and reading material viewable on handheld sized screens, and portable as load-and-go study material.

Brad Myer's Pebbles (Myer, 2001) project at CMU is developing toolkits to program handhelds to remotely control PC applications, and other devices in the environment. Handhelds can be linked directly to devices through their serial ports and are often used as tethered controllers for their touch screen interface, or, in the case of a cell phone, telephone interface. The IrDA port can also be programmed to act as a remote control, although most have a range of less than a meter. The different possibilities for connectivity and control provide limitless opportunity for fun and exciting projects in Senior Project classes for both the CS and CE curricula.

3. Classroom Experience

Much of the literature on handhelds in education focuses on the use of handhelds in Kindergarten through 12th grade education. For example, a recent study by SRI found that the use of PDAs in the classroom increased student motivation, even if they were used only for fairly standard tasks such as word processing (Tatar, 2003). In this paper we argue that the same motivating effect of handhelds can be extended to programming tasks in the computer science curriculum.

Several universities have offered explicit PDA programming courses, usually aimed at the Palm Pilot. Examples include Bryn Mawr's PDA Programming class (Mercuri, 2004), San Jose State University's Palm Programming class (Pollet, 2004), and Albright College's Handheld Wireless Technology class (Albright, 2004).

Palm Pilots were used to teach the Microprocessor course at Stanford University (Anderson, 2004). In that course, students used CodeWarrior for the assembly programming, rather than native tools on the device.

At Armstrong Atlantic State University, Mark Burge has incorporated some J2ME programming in the honors version of their Introduction to Programming Principles course (Burge, 2003, Burge, 2004a). Dr. Burge recently received an NSF grant to incorporate handheld and ubiquitous computing into more of the core computer science curriculum (Burge, 2004b). He already teaches several graduate courses on handhelds and pervasive computing, including one using Java Smartcards.

At the West Point Military Academy, students and faculty were issued PDAs, but only given basic instruction in their use. So far programming the PDAs has only shown up in the Senior Project course (Alford and Ruocco, 2001). PDAs are showing up in Senior Project courses at other institutions as well. The University of Connecticut's Computer Science & Engineering Design Laboratory is just one example (Demurjian, 2004).

Handheld PDAs and cell phones are the subject of any number of graduate research projects, as any HUC/UbiComp program will demonstrate. Here we cite just a few student projects to give some ideas about types of applications that are specific to a local context, and that have the potential to benefit the local community. Graduate students at the University of Genoa developed a PDA-based game that teaches other students about the art and history of Venice (Bellotti, et al., 2003). Graduate students at Berkeley created a GSM equipped cell phone application to support carpooling to reach needed services and local events for low-income residents in the California Central Valley (Kam, et al., 2004). Graduate students at UCSC are using GPS-enabled cell phones with agent systems to provide smart location-aware travel assistants for the Bay Area corridor (Griss and Yin, 2004).

4. Conclusion

Experience with the use of PDAs and cell phones in the undergraduate computer science curriculum is preliminary at this point. While it has been used to some extent in some courses, it has been treated more as a novelty – a tool, the use of which requires special instruction as an aside to the main material in the curriculum. But the potential is there. Some day, programmable handheld and ubiquitous devices will be essential tools that we carry with us at all times. The computer scientists of tomorrow could benefit from experience with these devices today.

Useful tools for programming to, and on cell phones and PDAs exist. Most are free or available to educators at low cost. The major hardware, software, and service vendors are all eager to see our students using these devices, and creating applications for them. The popularity of these tools in the classroom will no doubt grow as camera equipped, Windows, Palm, and Linux powered smart phones become standard accessories for students, and for non-students, alike.

References

- Sun Corporation. (2004). The Source for Developers: Java Technology, Java 2 Platform, Micro Edition (J2ME), <http://java.sun.com/j2me/>, 4/30/2004. (date accessed)
- Albright College (2004). Albright College Student Catalog, CSC 385, Handheld Wireless Technology, <http://www.albright.edu/catalog/compsci.html>, 4/30/2004. (date accessed)
- Alford, K.L. and Ruocco, A.S. (2001). "Integrating Personal Digital Assistants (PDAs) Into a Computer Science Curriculum," IEEE Frontiers in Education, Reno, Nevada, October 10-13, 2001; <http://fie.engrng.pitt.edu/fie2001/papers/1064.pdf>, 4/30/2004. (date accessed)
- Anderson, D. (2004). CS110, <http://cse.stanford.edu/class/cs110/>, 4/30/2004. (date accessed)
- Bayer, F. (2004). LispMe Home Page, <http://www.lispme.de/lispme/>, 4/30/2004. (date accessed)
- Bellotti, F., et al. (2003). "VeGame: Exploring Art and History in Venice," *IEEE Computer*, 36(9), September 2003, pp. 48-55.
- Burge, M.J. (2003). "A handheld and ubiquitous computing curriculum," in Proc. 41st Annual ACM Southeast Conference, pages 22-24, Savannah, GA, 2003. ACM.
- Burge, M. (2004a). <http://vision.armstrong.edu/burge/>, 4/30/2004. (date accessed)
- Burge, M. (2004b) "Pervasive Computing in the Undergraduate Curriculum," SIGCSE '04, Norfolk, Virginia, March 3-7, 2004; <http://db.grinnell.edu/sigcse/sigcse2004/viewAcceptedSession.asp?sessionType=SpecialSession&sessionNumber=16>, 4/30/2004. (date accessed)
- Collins, J. (2004). Pippy: Python for the Palm, <http://pippy.sourceforge.net/>, 4/30/2004. (date accessed)

- Demurjian, S.A., (2004). CSE293 Computer Science & Engineering Design Laboratory, PDA and/or Cell Phone Projects, <http://www.engr.uconn.edu/~steve/Cse293/cse293pilotphonespr03.pdf>, 4/30/2004. (date accessed)
- Fromme, O. (2004). Olli's Zaurus Pages, <http://www.secnex.de/~olli/Zaurus/>, 4/30/2004. (date accessed)
- Griss, M. and Yin, M. (2004). "SCATEAgent: Context-Aware Software Agents for Multi-Modal Travel," submitted to AAMAS workshop on transport, 2004
- Guillot, P. (2004). Bienvenue sur PP Compiler, <http://ppcompiler.free.fr>, 4/30/2004. (date accessed)
- Kam, M., et al. (2004). "Location-Based Services for Low-Income Communities in the California Central Valley," Berkeley EECS Annual Research Symposium (BEARS), Berkeley, California, February 27, 2004; <http://www.eecs.berkeley.edu/BEARS/STARS/final/kam.pdf>, 4/30/2004. (date accessed)
- Keuchel R. (2004). General Paranoyaxc Software, <http://www.rainer-keuchel.de/software.html>, 4/30/2004
- Lawrence, R. (2004). OnBoard Suite: A C language integrated development environment that runs right on your Palm PDA and compiles native Palm executables, <http://onboardc.sourceforge.net/>, 4/30/2004. (date accessed)
- Mercuri, R. (2004). CS 246: Programming Paradigms PDA Programming, <http://mainline.brynmawr.edu/%7Ermercuri/Fall02/CS246syl.html>, 4/30/2004. (date accessed)
- Microsoft Corporation. (2004). Mobile and Embedded Application Development Center, <http://msdn.microsoft.com/mobility/>, 4/30/2004. (date accessed)
- Myers, B. (2001). "Using Hand-Held Devices and PCs Together," *Comm. ACM*, Nov. 2001, pp. 34-41; <http://www-2.cs.cmu.edu/~pebbles/>, 4/30/2004. (date accessed)
- Mykland, R. (2004). Palm OS Programming from the Ground Up, Second Edition, Osborne Distributed Products, 2004
- Palm Corporation. (2004). PalmSource: Palm OS: The software that powers the world's most popular handhelds, <http://www.palmsource.com>, 4/30/2004. (date accessed)
- Plkr.org (2004). Plucker, <http://www.plkr.org/>, 4/30/2004. (date accessed)
- Pollett, C. (2004). CS185C Spring 2004 PDA Programming, <http://www.cs.sjsu.edu/faculty/pollett/185c.1.04s/>, 4/30/2004. (date accessed)
- Softfield Technology Corporation (2004). VR3 Linux PDA Specifications, <http://www.softfield.com/vr3.html>, 4/30/2004. (date accessed)
- Tatar, D. et al. (2003). "Handhelds Go to School: Lessons Learned," *IEEE Computer*, 36(9), September 2003, pp. 30-37.