Problem-Based Learning Using Mobile Devices

Christian Wattinger, Duc Phuong Nguyen, Peter Fornaro, Martin Guggisberg, Tibor Gyalog, and Helmar Burkhart
Computer Science Department, University of Basel, 50 Klingelbergstr, Basel, Switzerland
Phuong.Nguyen@unibas.ch

Abstract

Small handheld devices such as PDAs and smart phones become more and more popular. Thus, we are seeing a growing number of projects using mobile devices for educational purposes. Due to the limitations of screen size, CPU performance, and memory size, software development for mobile devices is challenging. More critical argument is about the lack of pedagogic and didactic concepts on the usage of mobile devices in education. In this paper, we present a system using mobile devices which supports research experiences for students in a laboratory context. The system allows students to efficiently monitor and control their scientific experiments at anytime, from anywhere. We describe several usage scenarios in the area of nanoscience studies, where the remote control of microscopes is mandatory. We present our system architecture and describe the implementation based on open-source software.

1. Introduction

Few years ago, the usage of mobile devices was very limited. For example, a mobile phone provided only voice communication and short message service (SMS), and a PDA only had simple features like calendar, to-do-list, and address book. The situation has significantly changed with the introduction of General Packet Radio System (GPRS) networks [1] and will improve further with the eventual advent of the Universal Mobile Telecommunications System (UMTS) standard [2]. These standards will allow always-on connection to the internet, without cumbersome configuration steps, slow and expensive connections as still seen today. A second improvement of current mobile devices is enhanced programmability such as through a built-in Java Virtual machine. Users may download small applications to their devices.

Despite this progress, most of today's mobile devices are still characterized by limited memory and modest CPU performance. The small screen size seems to be an inherent attribute of mobile hardware. Therefore, not all kinds of applications are appropriate for mobile devices. New technologies and approaches need to be developed to overcome device limitations [3], [4], [5]. It is suggested that mobile learning applications would be better suited with goal-directed tasks; they should be easily adapted to small screens and performed within a variety of user situations [6].

Having taken into consideration these issues, we decided to use problem-based learning (PBL) as driving force for our system. PBL is a technique that is used to improve students' achievement and motivation. Instead of passively learning a topic in a lecture, in PBL approach, students work in groups to study complex, real-world problems. The instructor guides them, adjusts their works and helps them analyze the results. In this way, students construct their own understanding by relating concrete experience with existing knowledge. Scientific experiment is good candidate for PBL. In this paper, the term scientific experiment refers to two types of experiments:

- Experiments with real devices such as atomic force microscope and medical diagnosis devices.
- Virtual experiments by means of computer-based simulations.

We designed a system that uses mobile devices for monitoring and controlling of scientific experiments. The system can tolerate the limitations of mobile devices because it does not require much user input and the users don't have to read a lot of text. It is also easy to learn how to use the software through a common user interfaces applicable to a wide range of real experiments and simulations. Moreover, the middleware layer optimizes the data according to the client's capability such as screen size and sends the data to the client in a bandwidth-saving manner.

0-7695-2632-2/06 $20.00 © 2006 IEEE

Complex scientific experiments can run for hours or days. Our software helps students to use their time efficiently. They don't have to stay in the lab all the time but still be able to monitor and control their experiments. We see several advantages such as:

• Monitoring an experiment from everywhere anytime. Researchers and students might not always have access to a desktop PC (e.g.: they leave the lab or even the campus). Laptops sometimes are not convenient to carry with. PDAs and mobile phones however can be easily carried with and are always ready to be used. This means that it is possible to observe an experiment while users are on the move (for example: in train or in library).

• Steering of remote experiments. When errors occur, users don't have to come back to their lab to solve the problem. They can use their mobile device to restart experiments, adjust the experiment parameters, and so on. This is especially useful for long-time experiments.

• Receiving instant notification from collaborators, tutors, and experts. Fast and simple notifications can be sent through a single button press. This includes the possibility to let other users immediately have a look at the measurement regardless of their location, supposed they have an instance of the client application on their mobile device.

• Self-tests and automated measurements. This may result in automatic alerting of a human supervisor for example in case of malfunction. A human supervisor has only to setup the experiment and is ready to react when receiving an alert (via email or SMS).

2. Related Work

There are several kinds of software for handheld devices that are used in educational activities. Examples of such applications are:

- Class response system [7], [8]: the teacher poses short questions or multiple-choice questions, students use their handheld devices to send back answers. The software collects and aggregates student responses and displays them in a coherent form (e.g.: histogram or bar chart). While these systems help teachers to get feedback of student's progress, they do not support self-learning.

- Software for reading course-related materials or making personal annotation: software found on handheld devices includes web browser, email/SMS, and eBook reader [9], [10], [11] as well as education-specific software [12]. Due to the small screen size and limited user input facilities of mobile devices, this kind of application is useful only with excerpts, short reading, and short annotation. Many users found that it is difficult to read long texts on mobile devices because they have to do a lot of scrolling and paging.

- In [13], mobile phones are used to manage online course content. Several actions can be done through mobile devices such as open/close web courses, add/remove some contents, and add homework answers. This kind of application is not considered a learning tool but an administration tool.

- Systems like [14], [15] use mobile devices for collaborative tasks in classroom such as discussion, chat, etc.

3. User Scenarios

In this section, we present some examples related to nanoscience. These examples are taken from the daily tasks of a group of physicists at the University of Basel. Nanoscience is a hot topic in biology, chemistry, and physics today [16-17]. That's why we chose one of the most important nanoscience instruments, the atomic force microscope (AFM) [18-20], as a prototype experiment for the remote control integration with the mobile thin client. Put in a nutshell, an AFM is a scanning type microscope: a micro-fabricated probe consisting of a tiny tip attached to a cantilever-shaped spring is scanned over a sample surface and the interacting force is recorded. The data yields a topography image of the surface, which is built up line by line as the measurement continues.

To illustrate the benefits of using our system, we assume a team consisting of student Alice, tutor Bob, and lecturer Charly.

Scenario 1: Virtual experiments and mobile learning

Alice is on the train going to the University of Nanoworld. Yesterday, she ran an experiment in the main server of the University because it is a complex simulation that requires a lot of computational resources. Even using the powerful server, it takes hours to run the simulation. Alice takes out her PDA, connects to the experiment server to check the ongoing experiment. From the measurement data visualized, she gets an overview of the result. Using the PDA, Alice also sees all the experiment parameters. She uses them to prepare for a short presentation in her seminar next week. The mobile device helps Alice to make more effective use of her time.

Scenario 2: Collaborative working

While Alice is working in the lab on the instrument, the device shows an error message. She wants to consult Bob about the problem. She presses the 'notify' button and selects Bob from a pop-up list. A short message is sent to Bob's mobile phone. Bob, who works in another building, immediately gets a view of the ongoing measurement as well as the associated experimental parameters. After checking the camera view of the sensor, also transmitted to the mobile device, he solves Alice's problem by repositioning the microscope's sensor on a more suitable spot.

Scenario 3: Instant Notification

Alice has observed an extraordinary experiment result. She decides to show the measurement to Charly, who is attending a conference. Alerted by an SMS, Charly connects his PDA to the running experiment. Seeing the result and the input parameters, he judges the measurement as of remarkable result and decides to show it as a live measurement in his conference talk on the following day.

Scenario 4: Accessible anywhere, anytime

Alice is having lunch and meanwhile in the lab, the atomic force microscope records data. Before getting her coffee, she quickly checks her mobile phone to see if the experiment goes right. Using the mobile phone, she could restart the measurement or adjust some parameters to improve the result. In case the experiment has a serious problem, she comes back to the lab to save precious measurement time.

Figure 1 shows several user interfaces of our software running on a Palm device.

Figure 1. Application User Interface

a) View of an observed surface with atomic resolution using the friction force mode.

b) Interactive list of gauges to view and adjust experiment parameters.

c) Web cam view of a microscope

4. System architecture

Overview

Figure 2 shows the system architecture, including the experiment server (running the experiment), the middleware component, and mobile clients. The experiment server is used to control real devices (e.g.: a microscope or a medical diagnosis device) or to run experiment simulations. We used the virtual interactive experiments framework (VEXP) [21-22] to run an Atomic Force Microscope simulator. The middleware coordinates the data flow between the experiment servers and clients. It sends client requests (commands) to the experiment server and receives the measurement data from the experiment server, converts the data into optimal images (according to device type) and sends the images to the mobile client. As mentioned above, mobile devices have limited memory and CPU power. Therefore, it is advisable to use them as thin clients. The clients are used only for simple tasks such as presenting data and gathering user input.

System components

Data from an experiment usually comes in two continuous and persistent streams:

- The command stream is a bidirectional stream, which transmits commands to the experiment and receives status messages from the experiment. In our case, a simple command language consisting of key-value pairs is used to steer the experiment. Examples of commands and responses for a position change and request of the measurement range looks like:

command = goto1
command = get name=scansize
RAFM>command=status name=scansize

The raw data stream flows from the experiment to the middleware layer and contains encoded measurement data. In the AFM case, the data rate is about 100 kbps (at high measurement speeds with many measurement channels). The raw data has to be decoded, filtered and processed in order to yield a displayable image. Because of limited computing performance and bandwidth we do not connect the two persistent streams to the mobile device. Instead a middleware component is connected to the experiment's data streams.

The middleware does the data processing burden and send the result as optimized images (according to the device capability) to the mobile device. The mobile client presents the visualized data and accepts control commands from the user.

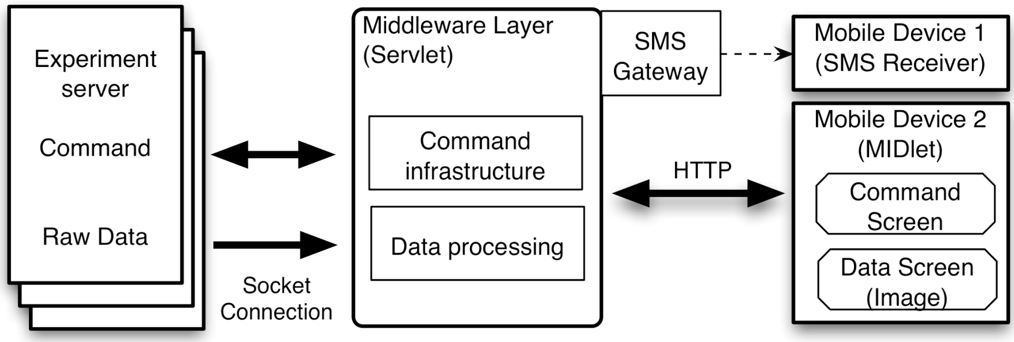

Figure 2. System architecture

The experiment server sends a stream of experiment data and receives commands from the middleware. The middleware coordinates the dataflow between mobile, web-based clients and the experiment server.

Mobile client

The mobile client has four modules:

1. The monitoring module visualizes the measurement data.

2. The command module allows the user to start, stop and pause the experiment as well as to change the current experiment parameters.

3. The notification module allows the user to inform other mobile users about the experiment. Several types of messages such as alert, special interest, or help requests can be sent to a specified list of receivers.

4. The viewer module is used to toggle between views of attached web cameras.

Middleware component

The client connects to the real experiment or simulation via the middleware layer. Clients request visualized measurement data via HTTP requests. To save expensive GPRS bandwidth, only the updated parts of the measurement data are sent. The measurement data in scanning type experiments (e.g.: scanning probe microscope experiments) is generated line by line. On stable network connections, there are many requests per unit time for very thin slices. That yields very good real-time visualization of the experiment. When the GPRS signal strength is bad, the delay between experiments can be a few seconds. If the delay time is longer than the scan duration for the complete image (around 60 s), the client gets the image in one chunk.

A request contains several parameters such as start and stop line of the actual slice, scaling factor and device type. Parameters are sent to the middleware servlet as URL encoded name-value pairs. In the same way, control commands for the experiment server are sent. The measurement data is received at the mobile client as a HTTP response (content type image/png). Image parameters and command responses from the experiment are included in the HTTP response as custom headers (e.g.: startline: 12, scanrange: 50). According to the image parameters (startline and stopline), the image slice is superposed to the image shown to the user. The user gets the impression of watching an evolving scan image as if he sat directly in front of the microscope.

The experiment continuously streams out raw measurement data on many channels. This image data flow can be as high as a few hundred kbps depending on the scan speed and the number of channels. As middleware and experiment server are usually located in the same LAN, only the transfer speed of the LAN connection sets a limit to the raw data stream.

To reduce the workload on the mobile client, only the data that the user wants to see at a given time is sent. To be able to react immediately to the demands of the user, it has to keep an up-to-date set of the uncompressed raw data matrix. The middleware component includes an image generator which creates PNG type images from the raw data (scaled to the client screen size). So measurement data is sent to the wireless client in a bandwidth- and transmission-cost-saving manner.

The middleware component is a Java servlet that provides scalability for many simultaneous clients. The J2EE technology enables a rapid implementation of an additional web application to control the experiment server. Web browsers of all computer platforms can display the HTML view of the servlet.

4. Conclusions and Outlook

We have introduced advanced software for problem-based learning in a laboratory context. Our software motivates students by allowing them to work collaboratively on similar scientific problems that researchers have in their daily routine. The system can be used by a variety of mobile devices and in a wide range of applications such as physical experiments, medical experiments, and long run-time calculations.

As we aim towards generalization of the mobile monitoring component, we envisage an abstract middleware layer and mobile client, which are configured according to the desired experiment. Most experiments will benefit from data visualization like our AFM example: views from related web cams, feature buttons and menus to check parameters, trigger actions and send notifications. The optimal user interface will look different for every experiment. One experiment can have more than one user interface, for example, one for administrators, one for regular users and one for novice users. An experiment descriptor file will contain an unambiguous description of all menus, views, selectors, and buttons, as well as of their functions. The servlet and the mobile client will access this descriptor file and configure the according filter and preprocessing as well as the GUI elements.

5. References

[1] Christian Bettstetter, Hans-Joerg Vogel, and Joerg Eberspacher, "GSM phase 2+ general packet radio service GPRS: Architecture, protocols, and air interface", IEEE Communication Surveys, 2(3), 1999.

[2] J. Rapeli, "UMTS: Targets, system concept, and standardization in a global framework," IEEE Personal Communications, Vol. 2, No. 1, pp. 20-28, 1995.

[3] Hao Liu, Xing Xie, Wei-Ying Ma, Hong-Jiang Zhang, "Automatic Browsing of Large Pictures on Mobile Devices", 11th ACM International Conference on Multimedia, Nov. 2003.

[4] O. de Bruijn, R. Spence, and M.Y Chong, "RSVP browser: Web browsing on small screen devices", Personal Ubiquitous Computing, 6(4):245–252, 2002.

[5] Yu Chen, Wei-Ying Ma, Hong-Jiang Zhang. "Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices". ACM Press, 2003.

[6] Maria Uther, "Mobile Internet Usability: What Can 'Mobile Learning' Learn from the Past?", IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'02).

[7] Roschelle, J., & Pea, R., "A walk on the WILD side: How wireless handhelds may change computer-supported collaborative learning", International Journal of Cognition and Technology, 1(1), 145-168, 2002

[8] Carl D. Fulp and Errin W.Fulp. "A Wireless Handheld System for Interactive Multimedia-Enhance Instruction", 32nd ASEE/IEEE Frontiers in Education Conference, 2002.

[9] Catherine C. Marshall, Christine Ruotolo. "Reading in the Small: A study of Reading on Small Form Factor Devices", Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, ACM Press, 2002.

[10] J. Waycott and A. Kukulska-Hulme, "Student's experiences with PDAs for reading course materials", Pers Ubiquituos Computing, 7:30-43, Springer-Verlag London, 2003.

[11] Patricia Thornton, Chris Houser. "Using Mobile Phones in Education," Proceedings of the 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education, 2004.

[12] Niels Pinkwart, Heinz Ulrich Hoppe, Marcelo Milrad, Juan Perez, "Educational scenarios for cooperative use of Personal Digital Assistants", J. Comp. Assisted Learning 19(3): 383-391 (2003).

[13] Ryosuke Komatsu, Jianhua Ma, Qun Jin, "A Multi-Agent System for Online Course Content Management", AINA, Volume 1 p. 183, 2004.

[14] Lars Bollen, Sabrina Eimler, H. Ulrich Hoppe. "The Use of Mobile Computing to Support SMS Dialogues and Classroom Discussions in a Literature Course," Proceedings of the Fourth IEEE International Conference on Advanced Learning Technologies, 2004.

[15] Pauliina Seppala, Harri Alamaki. "Mobile Learning and Mobility in Teacher Training", Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education, 2002.

[16] Engel, Hans-Andreas, Loss, Daniel Fermionic, "Bell-State Analyzer for Spin Qubits", Science, 309: 586-588, 2005.

[17] Mühlschlegel, P., Eisler, H.-J., Martin, O. J. F., Hecht, B., Pohl, D. W. „Resonant Optical Antennas", Science 308: 1607-1609, 2005.

[18] G. Binnig, H. Rohrer, Ch. Gerber, and E. Weibel, "7x7 reconstruction on si(111) resolved in real space", Phys. Rev. Lett., 50(2):120–123, 1983.

[19] F.J. Giessibl, "Atomic resolution of silicon(111)7x7 by atomic force microscopy through repulsive and attractive forces", Science, 267(68):1451–1455, 1995.

[20] A. Socoliuc, R. Bennewitz, E. Gnecco, and E. Meyer, "Transition from Stick-Slip to Continuous Sliding in Atomic Friction: Entering a New Regime of Ultralow Friction", Phys. Rev. Lett. 92, 134301, 2004.

[21] Virtual interactive Experiments (VEXP) framework: http://vexp.nano-world.org/ (Last visited on 10/04/2006)

[22] M. Guggisberg, P. Fornaro, T. Gyalog and H. Burkhart, "An Interdisciplinary Virtual Laboratory on Nanoscience", Future Generation Computer Systems 19, 133-141, Elsevier, 2003.