Communication Ubiquity Enables Ubiquitous Control

Proposal for a Human-Computer Interaction Consortium 2002 Presentation

Brad A. Myers and Jeffrey Nichols

Human Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3891
(412) 268-5150
FAX: (412) 268-1266
bam+@cs.cmu.edu
http://www.cs.cmu.edu/~bam

November 27, 2001

Short Abstract:

Handheld devices, such as cell-phones and PDAs, can be used for more than just communication. What happens when every person has a handheld that ubiquitously communicates with every appliance in the environment? As part of the Pebbles project, we are exploring technologies that allow handhelds to *monitor and control* appliances.

Extended Abstract:

Everyday appliances, including telephones, copiers, home stereos, light switches, etc., increasingly contain embedded computers which enable greater functionality. If the interfaces to these appliances were easy to use, people might benefit from these new functions. Unfortunately, it is rare to find a well-designed appliance interface [Brouwer-Janse 1992].

We are investigating how an intelligent hand-held *Personal Universal Controller* (PUC) can be used to make appliances easier to control as part of the Pebbles research project [Myers 2001]. The concept is that people can use their own handheld to control all appliances and devices in their environment, such as the lights, a photocopier in an office, a machine tool in a factory, a VCR at home, a piece of test equipment in the field, etc. The appliance sends the handheld a description of its control parameters. The handheld uses this information to *automatically* create an appropriate control panel, taking into account the properties of the handheld (the display type and input techniques available) and the properties of the user (what language is preferred, whether the user is left or right handed, how big the buttons should be based on whether the user prefers using a finger or a stylus). The user can then control the appliance using the handheld. Unlike today's universal remotes, the PUC will be self-programming and use two-way communication: it will create control panels for new devices without user intervention, and it will receive feedback from the appliances to help keep the user informed as to the device state.

We believe that the PUC can address many of the difficult problems inherent in ubiquitous technologies. Here are some examples:

- As users move through their environment, they may not be aware of the services available to them or the appliances that they may be able to control. **A personal controller gives the user a means to discover what is available around them.** The service discovery layer of the Bluetooth protocol [Bluetooth 1999] or a similar standard could be used to locate the controllable services and appliances within range. These could be presented to the user in a list on the controller. A map of relative appliance and service locations could also be generated, where appropriate, if a location service [Bahl 2000] was available to the controller unit. This enables users to, for example, quickly locate the nearest public video phone when they need to make a video conference call. They could also quickly see what electronic amenities are available in an unfamiliar hotel room without thumbing through pages of hotel documentation.

- While each user will have a space of appliances that they are familiar with, it is certain that they will encounter unfamiliar appliances when they enter a new environment. This could be in a hotel room or at a customer's office. Although many of the appliances found in these new spaces will offer the same functionality as familiar appliances, they may have user interfaces that are inconsistent with the user's previous experience. For example, many people have difficulties using alarm clocks in hotel rooms. Everyone knows what the device is for, but there is such variation across different models that most people call the front desk for a wake-up call instead of even trying to set the alarm clock. Similar stories apply to copy machines, home stereos, and voicemail systems. A PUC solves this problem by providing a manufacturer-independent interface to all devices. The same interface that the user is familiar with to set the alarm clock at home could be used on the clock in the hotel room, and even to set the time for a recording on a VCR. **The PUC can exploit previously generated interfaces to maintain consistency and improve usability.**
- **No user is the same, so a ubiquitous interface must be sufficiently customizable to work for a wide range of users.** The PUC handles this by allowing the user to influence how interfaces are generated for unfamiliar devices. This might involve moving controls to accommodate a left-handed user or using larger labels for older users with poor eyesight. A direct manipulation interface would also be provided for those users who want to customize interface elements. Users could drag elements out of a current interface and lay them out manually on the controller screen. Users could even share their screens as is popular for devices such as the Philips Pronto Interactive Remote Control [Philips 2001, RemoteCentral 2001].
- No appliance or service exists in a world of its own. Users may find that they commonly deal with several particular appliances at one time and would benefit from combining some functionality from each into a single "custom" interface. The controller unit provides a convenient location for such an interface to be created and used. The direct manipulation method mentioned above does not need to be confined to creating new screens for a single appliance. **Commonly used functions from multiple appliances in the environment could be combined to create control panels.** The user might put controls for their overhead lights on the same panel as the television power button, for example. The user would also be allowed to create macros, sets of commands that might control several devices in a single stream. Macros could be assigned to controls on a custom interface screen. A button to turn on the cable box might also turn on the TV and change it to channel 3. Another example is that the user might create several "mood" buttons, for example one for setting the lights at different levels and playing different selections on the CD player.
- If users are listening to their stereo and are then interrupted by the telephone, they might like to turn down the stereo volume before they begin speaking on the phone. They may not always want the volume turned down however, and could be annoyed by an automatic change. **The controller unit allows the user to stay in control of their environment.** When the phone rings, the user picks up their controller which is showing a phone interface with a volume control and mute button for the stereo overlaid in one corner. The user hits the mute button before pressing the control that picks up the phone. These kinds of interactions are easily enabled because the controller unit can establish a two-way connection with all the devices and services in the current environment. It can perform a query to discover what is in use and present appropriate controls to a user when the state of the environment changes.

There are many significant research issues involved in bringing this vision to life, and we are in the process of investigating many of them. Our initial focus is on interface generation and the specification language that supports it. We began our research by creating hand-designed interfaces for an AIWA shelf stereo system and an AT&T office telephone/answering machine, targeted both at the Palm and PocketPC hand-held computers. The purpose of the hand-designed interfaces is two-fold:

1) To verify that better remote control interfaces can be created on a hand-held.

2) To understand what functional knowledge about an appliance will be needed to construct a high-quality remote control interface.

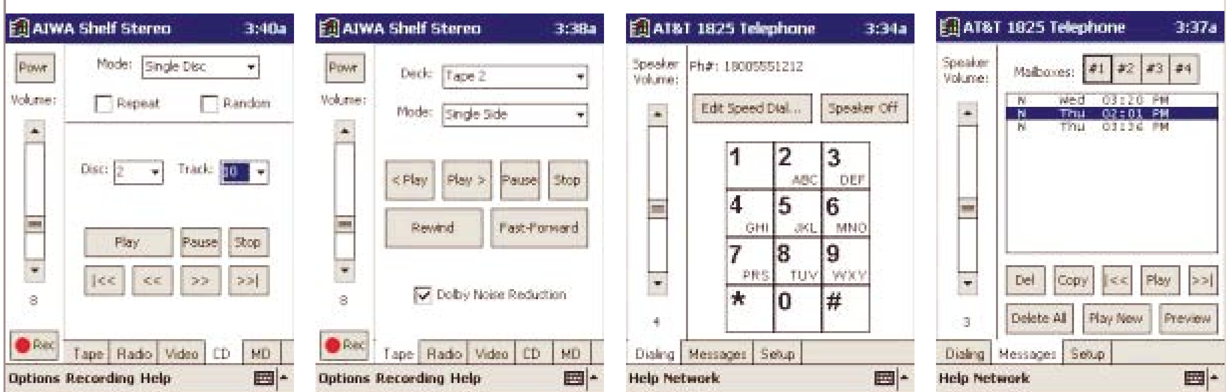
We performed two user studies to test the first point. Results showed that users made about 1/5 the errors and took about 1/2 the time to perform complex tasks using the handheld interfaces compared to the actual manufacturer's interfaces for the stereo and telephone [Nichols 2001]. We have also gained knowledge that is needed to create high-quality remote control interfaces. For example, we have noticed that the layout problem for many remote control interfaces can be greatly reduced by grouping together related items. These groups can often be reduced to having a small number of items, which are then easy to place on the screen relative to each other. Furthermore, these groupings can often be inferred from knowledge about when a function can be used. For example, in the stereo interface shown in the figure below, it is possible to infer that Tape and CD functions should be in separate panels because their functions cannot be used at the same time.

Now, we are using the knowledge gained from evaluating our hand-designed interfaces to create a specification language and an automatic generation and layout algorithm that will create similar interfaces. The challenge is to make the specification flexible enough to support a wide-range of form factors and interface modalities, while still providing high-quality user interfaces. As a part of this, we are working with Carnegie Mellon's Universal Speech Interface Group to investigate the possibilities of using our specification as the basis for speech interfaces. We are also exploring how to connect to real appliances. For example, we are collaborating with Premise Systems to look at using the PUC for home automation control. There is hope that standards such as Universal Plug-And-Play (uPnP), BlueTooth, Home Audio Video Interoperability (HAVi), or other companies' proprietary protocols will provide interactive two-way control of appliances in the future. Sony, for example, has set a goal of connecting most of their products to the network by 2005 [Sony 2001].

For the future, there are many technological, user interface, and cultural issues that will be important to study. There are questions of how to automatically generate remote control interfaces and make them adaptable to the user and their environment. Consistency and standardization have helped make it easier for users to understand interfaces in the past, but how will this change if every user has different, personalized interfaces to the same appliance? What is the proper balance between controls on the appliance itself, and controls on the PUC? We do not think that appliances should just become black boxes that can only be controlled from a handheld. There are also cultural issues to consider. How will appliances, homes, and offices change if most control has moved to handheld devices? We plan to study these and other issues over the next several years.

AIWA CX-NMT70 shelf stereo and its remote.

AT&T 1825 telephone/digital answering machine



Panels for the PocketPC to control the stereo and the telephone.

References

[Bahl 2000] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," *In Proceedings of the IEEE Infocom 2000*, Tel-Aviv, Israel, vol. 2, Mar. 2000, pp. 775-784. Adobe Acrobat (pdf)

[Bluetooth 1999] Bluetooth Consortium, *Specification of the Bluetooth System: Core*, Vol 1B, December 1999, http://www.bluetooth.com. Adobe Acrobat (pdf)

[Brouwer-Janse 1992] Brouwer-Janse, M.D., Bennett, R.W., Endo, T., van Nes, F.L., Strubbe, H.J., and Gentner, D.R. "Interfaces for consumer products: how to camouflage the computer?" in *CHI'1992: Human factors in computing systems*. 1992. Monterey, CA: pp. 287-290.

[Myers 2001] Myers, B.A., "Using Hand-Held Devices and PCs Together." *Communications of the ACM*, November, 2001. 44(11): pp. 34-41. Adobe Acrobat (pdf).

[Nichols 2001] Jeffrey W. Nichols. "Using Handhelds as Controls for Everyday Appliances: A Paper Prototype Study." ACM CHI'2001 Student Posters. Seattle, WA. March 31-April 5, 2001. pp. 443-444. Adobe Acrobat (pdf)

[Philips 2001] Royal Philips Corp. http://www.pronto.philips.com/, January 2001.

[RemoteCentral 2001] RemoteCentral.com, http://www.remotecentral.com/pronto/, January 2001.

[Sony 2001] Sony.com, http://www9.station.sony.com/sca/press/03292001.html, March 2001

Funding

This part of the Pebbles research project is funded by grants from DARPA, NSF, Microsoft and the Pittsburgh Digital Greenhouse, and equipment grants from Symbol Technologies, Palm, Hewlett-Packard, Lucent, IBM and SMART Technologies, Inc.

This research was performed in part in connection with contract number DAAD17-99-C-0061 with the U.S. Army Research Laboratory. The views and conclusions contained in this document are those of the authors and should not be interpreted as presenting the official policies or position, either expressed or implied, of the U.S. Army Research Laboratory or the U.S. Government unless so designated by other authorized documents. Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof. This research is also funded in part by the National Science Foundation under Grant No. IIS-0117658. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation.