smart-space researchers

An Interview with Kevin Mills and Alden Dima

by Kevin Mills

<kmills@nist.gov>

and Alden Dima

<alden.dima@nist.gov>

Editor's note: Rik Farrow conducted this interview electronically. Kevin Mills and Alden Dima, researchers at NIST, are working on Smart Spaces and Jini-enabled devices.

Rik: Would you describe in some detail what you are doing at NIST?

Kevin and Alden: In a recent year, 1997 I believe, 4.2 billion microprocessors shipped. Only about 165 million of these microprocessors shipped in desktop personal computers – the remainder shipped as microcontrollers embedded in devices, such as microwave ovens, automobiles, televisions, and climate-control systems. Almost all of these embedded microcontrollers consisted of standalone computing applications. Yet, when you look at where future technology is leading us, microprocessors are rapidly moving to integrate most functions of an entire computer system on the die. Similar miniaturization in radio technology is also leading toward integration of radios onto chips. Industry's Bluetooth initiative provides one example of that trend. Given these trends, we can foresee that the embedded-microprocessor market will become an embedded system-on-a-chip (SoC) market, and that many of those SoCs will contain integrated wireless radios. So, in some future year, we might see four billion SoCs shipping with the capability of networking with each other. Many of these SoCs will be embedded in our environment, and many will be embedded in mobile devices that we will carry around from place to place.

Several networking and software issues may inhibit the development of the marketplace for embedded SoCs. Primary among them is automatic configuration and dynamic discovery. We cannot rely on human system administrators to configure and administer four billion computer systems, many of which will move from place to place. In addition, most mobile devices will be designed to leverage services available on larger processors available through more traditional wired networking infrastructures. To leverage such services, embedded SoCs will have to discover their presence, gain authorized access to them, and then configure themselves to exploit them. Other issues arise because developers of software applications will have to use new models for programming applications in such dynamic environments. And, of course, user interfaces will become distributed across a range of devices that must cooperate to provide humans with a seamless and effective interface. All of these issues are challenging, and there are folks here in the NIST Information Technology Laboratory working on each of these topics. Alden and I are focusing most directly on issues related to dynamic discovery, automated configuration, and new programming models.

Many industry initiatives are beginning to develop technologies to address just these problems that will increasingly impede progress toward tomorrow's more dynamic, distributed, mobile environment. Consider for example, Jini, Universal Plug-and-Play, and Service Location Protocol. As our contribution to these efforts, we decided to assess the functionality being provided by existing industry proposals, and to develop user performance and resource utilization metrics that can be used to assess performance of the various technologies. After analyzing some nine different proposed dynamic discovery technologies, we are developing a functional taxonomy that can be used to compare and contrast specific proposals. Further, we plan to deploy and measure performance of several of the more popular proposed technologies.

Achieving this paradigm shift will require researchers to begin thinking of information objects as active, mobile, and context-aware.

To provide an assessment of the ability of these technologies to work with tomorrow's SoCs we have developed a prototype, which we call the Aroma adapter. Each Aroma adapter consists of three PCMCIA cards, which together emulate the type of SoCs we expect to be available within five years. To date, we have used this prototype to convert a computer-controllable video projector into the type of portable, wireless information appliance we expect to be available in future years. We then implemented Jini on the Aroma adapter, converting the video projector into a Jini device that can be discovered, dynamically configured, and accessed across a pico-cellular wireless network. This prototype has given us our first hint as to the functionality and performance available from one of today's leading dynamic discovery technologies.

Rik: In your USENIX paper "AirJava: Networking for Smart Spaces"(Workshop on Embedded Systems, 1999), you mention the "Three Challenges for Smart-Spaces Researchers." Could you briefly outline these three challenges? (I particularly found the concept of location-relevant data in section 2.2 far-reaching.)

Kevin and Alden: The movement toward heavy use of embedded and mobile wireless SoC technology, supported by clusters of computers connected to both the wired and wireless infrastructure, will create new challenges that researchers and industry must address before users will benefit. Three challenges seem particularly important. First, mobile users carrying cell phones and personal digital assistants into spaces rich with information appliances will be able to discover a wide range of interface devices, such as large-screen displays, voice- and vision-recognition systems, and controllable cameras. Individual users and groups of users might like to exploit these distributed devices to construct ad hoc human-information interfaces to seamlessly exploit the best modes of interaction for the best purposes. We might call such interfaces poly-device, polymodal (PDPM) interfaces. Some sort of distributed coordination bus will be required to compose such interfaces and to interact with information through them. Research will be busy working out the software issues and the human-interface issues presented by such environments.

A second challenge revolves around moving information for people. In today's model, either people carry their information with them, or they access that information remotely as they move from place to place. In either mode of operation, we must manage all our information for ourselves, even when much of the information we work with is context-dependent. For example, we typically attend meetings to conduct specific tasks. Before, after, and during these meetings we create information. Some of this information we retain personally, while other information is shared among the meeting attendees and others outside the group. Only a small fraction of this information is our own personal information.

Surely, as we move to the next meeting on the same subject we wish to have information from the last meeting available. At present, users must ensure that the necessary information is available at the right place and time. In the future, active information should be able to take on this responsibility. Imagine active information objects that can move, that can self-replicate, and that can communicate as a group. Such active information should be able to track the location, state, and trajectory of users, of object replicas, and of linked objects. In addition, active information objects should be able to plan the movement, replication, and transformation of information to serve the projected needs of its users, whether individuals or groups. Active information must also be able to implement consistency, access, and sharing policies among replicated and linked objects. Achieving this paradigm shift will require researchers to begin thinking of information objects as active, mobile, and context-aware.

A third challenge involves the transformation of information for presentation using knowledge about people, places, and devices. At present, network-based computing works because people carry in their minds a reasonably good model of cyberspace. We know that computers and printers can be found, and we know that information can be organized for storage on a disk. We know, but just barely, how to locate, download, configure, and execute various plug-ins to display information in specific formats or to convert information between formats. It appears that much of this knowledge is really rather routine, and that we could encode such knowledge into computers and networks, so that they could offload much of this routine work that we now require of the users of computers and networks. Using such knowledge, and the same heuristics that human users exploit, we could expect computers and networks to locate and compose devices and services in order to transform and display information in the best form for available output devices, personal preferences, and task effectiveness.

At least these three challenges must be overcome before users can be presented with effective, usable pervasive computing environments that leverage embedded and mobile, networked computer systems.

Rik: What about security? The idea of entering a Smart Space and being able to use a flat-screen display instead of my cell phone's puny LCD sounds great, but I can just imagine a confidential email flashing instead onto a large screen via the Aroma-enabled LCD projector. Are there any initiatives examining how to handle security?

Kevin and Alden: Good question. Security issues are currently unresolved, and along several dimensions. For example, given a set of embedded services, some of which should be made available to anyone who enters a space and others of which should be restricted to be used by particular individuals at particular times, how can access be controlled? This might be especially nettlesome because we can't expect every node to know about every conceivable user and to apply access restrictions on that basis. Probably work in the area of capability-based access-control mechanisms will help solve these issues. On the specific issue that you raised, users must be given mechanisms to set their own security policies (and other policies, for example about interruptibility). In my view, the composition of services and the routing of information to particular devices must be guided by policy mechanisms, some of which relate to user preferences, some of which relate to task, and some of which relate to context. These are all rich research areas that remain to be explored and solved.

The composition of services and the routing of information to particular devices must be guided by policy mechanisms, some of which relate to user preferences, some of which relate to task, and some of which relate to context. These are all rich research areas that remain to be explored and solved.

Rik: You mention a particular hardware implementation, called GUMPS. Could you describe what you are using in your prototypes, and perhaps provide URLs for sites selling something like this?

Kevin and Alden: We build two prototype adapters, both of which used components from various vendors. The components consist of four types: (1) Card PC, (2) PCMCIA wireless LAN interface card, (3) PCMCIA Flash RAM card, and (4) chassis and power supply. This approach to components is made possible by work on industry standards for PC cards for example: see *<http://www.pc-card.com/pccardstandard.htm>*. Card PCs are available from a number of vendors, such as Epson (*<http://www.epson-electronics.de/download/downcard.htm>*), Ampro (*<http://www.ampro.com/products/index.htm>*), and others. Flash RAM cards are available from a number of vendors, such as Kingston (*<http://www.amtron.com/price.htm#5>*), Synchrotech (*<http://www.synchrotech.com/products/ata-flash.html>*), and others. 802.11 wireless LAN cards available in two different technologies – frequency-hopping (*<http://www.proxim.com/>*) and direct sequence (*<http://wwwdb.lucent.com/bcs/>*) – spread spectrum, which are

I understand that a complete system with the same footprint as our assembled adapter can now be purchased off the shelf.

available from a number of vendors. The chassis and power supplies can also be acquired from multiple sources, typically the same sources as the PC cards. I understand from Alden that a complete system with the same footprint as our assembled adapter can now be purchased off the shelf.

Rik: A final question. You had a great list of URLs at the end of your USENIX paper (although none for yourself). Do you have a Web page with that list of links (and perhaps others)?

Kevin and Alden: We had such a Web site; however, Sun Microsystems requested that we change the name of our adapter technology from the name used in the paper to another name. We chose Aroma. As a consequence, we took our Web site off the air, and we haven't gotten a chance to update the site yet.