Editor: Vince Stanford ■ vince-stanford@users.sourceforge.net

Pervasive Computing Goes to Work: Interfacing to the Enterprise

Vince Stanford

EDITOR'S INTRODUCTION

Since Mark Weiser first conceptualized ubiquitous computing a decade ago, the notion that pervasive devices will revolutionize business practices—creating the vaunted paperless office—has become commonplace. Even so, offices have run on paper for centuries, if not millennia, and business leaders need real incentives to displace a venerable, cheap solution that works. As this installment of the Applications department illustrates, some companies are indeed finding appropriate pervasive solutions to displace paper.

For example, the McKesson Closed-Loop Distribution System uses pervasive personal digital assistants to manage billions of dollars in deliveries across North America. This system's deployment illuminates gritty issues that arise when deploying pervasive systems in the real world of loading docks, delivery trucks, and long, lonesome highways. Battery life, industrial ruggedness, small screen interfaces, and low-bandwidth intermittent connectivity all give pervasive application developers headaches. As we will see, real pervasive computing deployments must work without always-on wireless connectivity in situations where benign office conditions simply do not apply—and these systems have to pay their own way. —*Vince Stanford*

The paperless office is an idea whose time has come, and come, and come again. To see how pervasive computing applications might bring some substance to this dream, I spoke recently with key managers and technologists at McKesson Corporation (San Francisco), a healthcare supplier, service, and technology company with US$50 billion in sales last year, and also at AvantGo (Hayward, Calif.), a provider of mobile infrastructure software and services.

For the past several years, McKesson has used mobility middleware developed by AvantGo to deploy major supply chain applications with thousands of pervasive clients and multiple servers that replace existing paper-based tracking systems. According to McKesson's managers, their system greatly reduced errors and associated costs caused by redelivery or loss of valuable products, giving McKesson a solid return on its investment.

CONNECTING MOBILE WORKERS EVERYWHERE

In early 1997, as wireless connectivity and the first Palm Pilots became available, AvantGo's founders recognized the potential of fusing the key technologies that could effectively manage large forces of mobile workers with intermittently connected wireless environments. According to Felix Lin, AvantGo's vice chairman and cofounder, they cracked opened a Palm Pilot device and found double the processor power of the original Macintosh, as well as enough screen resolution for email and other common office functions. Clearly, applications far beyond the standard organizer and date book were possible. Other revolutionary technology trends were emerging contemporaneously, including the Internet, Web standards, and proliferating wireless services. (For a look at founding research done with intermittent connectivity, see the sidebar, "Coda and Bayou.")

According to Lin, who still guides his company's technical direction, industry and government have invested more than $400 billion in the Internet in recent years, with wireless carriers spending over $100 billion on spectrum alone—and more still on infrastructure. Customer relationship management, enterprise resource planning, analytics, and data warehousing software consume another $100 billion each year. So, founding a company to solve some of the numerous integration issues seemed like a good bet.

Lin and his colleagues thought the convergence of these technologies would open new business opportunities when lightweight, low-cost, consumer-grade, and wireless palmtops began to exploit the Internet. This convergence could make information technology—then already automating the corporate home office—available to mobile workers at reasonable cost. From the start, AvantGo's founding vision involved connecting the enterprise IT core to mobile workers everywhere.

1536-1268/02/$17.00 © 2002 IEEE

THIN CLIENTS VERSUS UNRELIABLE CONNECTIVITY

This vision would require a new class of enterprise middleware, improvements in client software on palmtops, and new data-management and transport capabilities to connect businesses directly to their mobile workers. There were numerous gaps to fill and technical challenges to meet. The palmtops themselves had different operating systems, memory limitations, displays, and connectivity options. Connectivity standards alone - involved alternative synchronization protocols and various wireless networking standards—TDMA, CDMA, 3G, 802.11, and Bluetooth among them. Also, most back-office business applications used proprietary application programming interfaces, query languages, and scripting languages. No single start-up company could solve problems this numerous and varied in its early development phases.

So the AvantGo founders predicated their technology development on the idea that, in the end, most back-office applications would move to the Web—or at least provide Web-based front-ends to connect them to the Internet. But for the mobile worker of the day, persistent, reliable connectivity of the wired Internet and thin-client, stateless Web transaction processing were only dreams. Bandwidth was low. Worse, it was intermittent and, worse still, most conventional Web servers and applications lost data when connections lapsed. Also, as mobile users know, interruptions were frequent, coverage poor, and data bandwidth miniscule in some areas.

According to Lin, the basic conflict between thick- and thin-client computing involves several straightforward factors. Thick clients, such as desktops, can store significant numbers of transactions, data entry pages, software, and data, but they need more storage and processor power than current palmtops offer. They are robust with respect to continuous connectivity because they replicate large databases for local processing during interruptions or for reducing communication overhead.

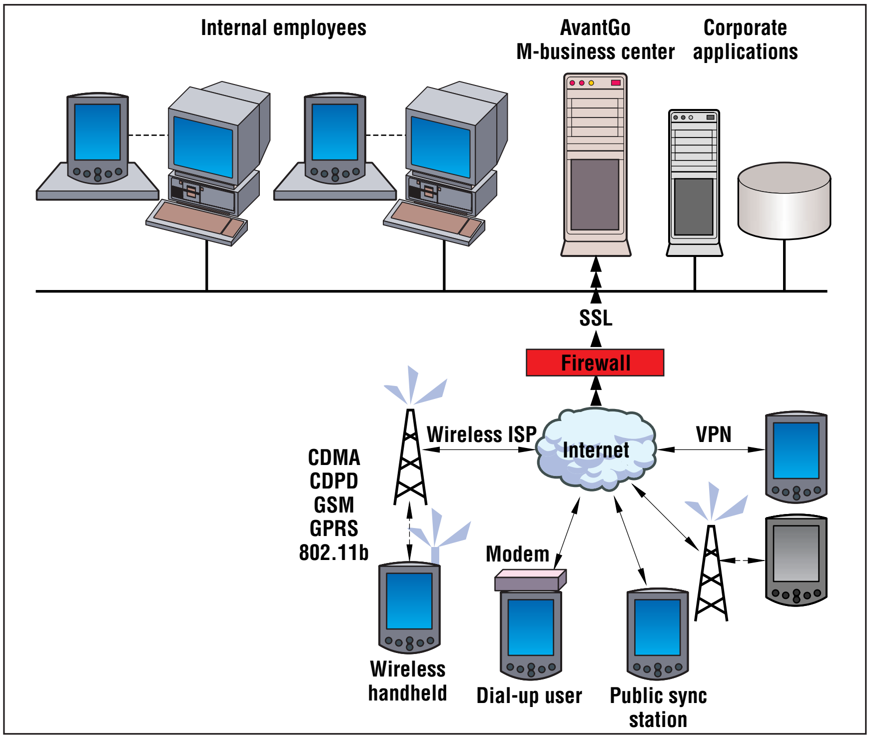

Figure 1. AvantGo middleware connection layer, illustrating middleware's role in mediating between mobile pervasive applications and enterprise databases.

This works well for department servers but not for thousands of clients, all intermittently offline, each with a replicated database. In such a case, proper reconciliation would be difficult, or even impossible, when resynchronizing with master databases. Figure 1 shows the intermediary role the AvantGo middleware system plays in a mobile pervasive computing application.

Thin clients, which are architected as stateless Web applications, require less local processing and storage compared to thick clients. Because they post transactions to servers immediately, they need not reconcile with master databases. However, they do require persistent connections for real-time updates and adequate bandwidth. Moreover, enterprise wireless data communications are more brittle than wireless voice communications, because human participants can pick up a conversation when they return from the point where it was interrupted. By contrast, interrupting a data transaction almost always results in data loss or corruption. Given the memory and processor constraints of palmtops, thin-client processing was really the only available option, although the lack of reliable wireless connectivity remained a major obstacle.

So, the AvantGo team needed to adapt the stateless and persistently connected Web to the mobile worker's intermittently connected, low-bandwidth environment. According to Lin, they devised an unintuitive solution, deciding to simulate an always-on connection even though one was not actually available. Their solution involved a distributed-processing architecture using platform-independent client software on mobile devices by employing intelligent caching, advanced session management, and Web page compression. With these, Web applications could function as if connected to the stateless server for a time, so the client could continue to present cached Web content and collect desired data on the palmtop. This way, mobile workers could enter data, send transactions to

CODA AND BAYOU

Contemporary pervasive middleware contains features designed to mitigate the effects of intermittent, low-bandwidth connectivity in wide-area networks. As we saw with the McKesson Closed Loop Delivery System, these features are crucial prerequisites to mobile deployments and will likely remain so. As with many excellent commercial products, earlier research proposed solutions to problems faced by products being fielded today.

For example, the Coda file system (www.coda.cs.cmu.edu) was developed at Carnegie Mellon University during the late 1980s and early 1990s. Although known today for robust-to-weak mobile file access and intermittent connectivity, mobility support was not an original design goal. Rather, mobile computing existed mostly in science fiction in early 1987 when Coda was first designed. The fault-tolerant design was actually driven by the frequent failures of the era's fragile network technology.

Thus Coda was designed for failure-resilient, scalable, and secure access to shared information across relatively fragile networks as well as application-specific conflict resolution. So it used mechanisms of server replication, disconnected operation, adaptive use of weak connectivity, isolation-only transactions, translucent caching, opportunistic exploitation of hardware surrogates, and application-specific conflict resolution. Coda was in turn influenced by other contemporary research efforts.

Another example, the Bayou system (www2.parc.com/csl/projects/bayou) developed at Xerox PARC starting in 1993, supported collaboration among intermittently connected users dispersed across the Internet or working on portable computers. Bayou assumed disconnected operation by mobile users as the norm, rather than the exception. Earlier replicated data algorithms, which maintained strong data consistency by atomically updating copies or cache invalidation, did not work well in frequently partitioned networks. Algorithms for partitioned networks either enforce locking of replicas or else fail to guarantee consistency and offer little support for resolving conflicts. Bayou used weak consistency techniques to manage replicas of shared calendars, email messages, databases, documents, and other artifacts central to collaboration. It included then-novel methods for application-specific conflict detection and per-write conflict resolution.

a specialized AvantGo server, and continue with their work as if connected.

Finally, the enterprise middleware layer maintains the necessary state to complete the transactions generated by mobile workers when out of communication. This architecture gracefully degrades to local palmtop capabilities during connectivity outages, simulating continuous connectivity as much as possible by using cached data. Later, when connectivity returns, the enterprise middleware uses the saved state of the interrupted transactions to match data to enterprise databases.

Thus, applications written to open Web standards could be deployed to mobile palmtops with minimal reprogramming. Developing mobile applications would simply involve using existing development tools and approaches and accounting for the smaller dimensions of mobile device displays. Advanced message queuing, distributed processing, and transaction management facilities could insulate applications from the challenges of mobile environments, which AvantGo middleware would transparently handle. This approach offers compelling advantages to making the back-office applications, which now support Web-based interfaces, available to the mobile worker. Figure 2 shows the AvantGo environment's major system components.

This architecture also supports critical security requirements of the mobile enter-

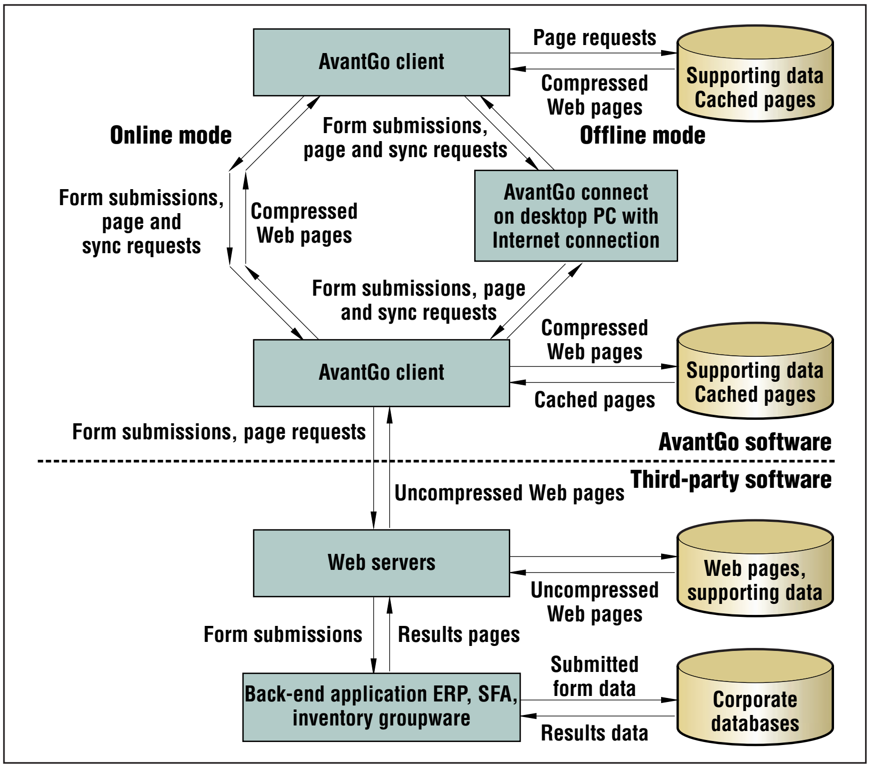

Figure 2. The AvantGo dynamic mobility architecture, illustrating the architectural features in the middleware letting applications function with intermittent connectivity.

prise. AvantGo middleware offers secure communication of information between mobile devices and enterprise applications inside corporate firewalls—even when information is transmitted over public wireless networks by using standard Secure Sockets Layer and Public Key Infrastructure. For large enterprises, the AvantGo server also provides a gateway that authenticates users and controls access to enterprise systems. It can also interoperate with active directories and NT domains, so that the enterprise administrators need not duplicate user accounts. (See the "IBM Pervasive Middleware" and "Pervasive Mobility Tools from Aether Systems" sidebars for a look at work being done elsewhere.)

IBM PERVASIVE MIDDLEWARE

IBM has developed a wide range of offerings in pervasive computing and is pursuing a vigorous research agenda in the area. It has long specialized in complex enterprise middleware, including operating systems, database management systems, communication systems, collaboration tools, and security management tools. As pervasive computing has developed, IBM has extended outward from its enterprise middleware core to the mobile pervasive devices, while companies like AvantGo work their way into the enterprise core from the mobile clients.

IBM has added products to support pervasive computing for wireless gateways, virtual private networks, secure transactions, Web commerce servers, pervasive device security, personal information management software, supply chain management, field service applications, and content transcoding tools. Also, both IBM and Oracle have begun offering database clients for pervasive operating systems. Web resources describing the IBM pervasive computing program are available at

- ***IBM Systems Journal*, vol. 39, no. 4, 1999:** www.research.ibm.com/journal/sj38-4.html
- **Discovering Devices in Home Networking:** www-3.ibm.com/pvc/tech/networking.shtml
- **Internet Transcoding for Universal Access:** www.research.ibm.com/networked_data_systems/transcoding
- **WebSphere Transcoding Publisher:** www-3.ibm.com/software/webservers/transcoding

REPLACING THE PAPER TRACKING SYSTEM

McKesson's Closed Loop Distribution project involves a supply-chain and delivery-management system using thousands of pervasive computing devices to replace the previous-generation paper invoice system. According to project manager Jesse Bork, this business faces ongoing pressure on profit margins, requiring hard returns on investment—rather than assuming that more technology is better technology. Cost savings are the project's primary success criteria, according to Bork: "We have experienced significant cost savings with the proof-of-delivery system every year."

Other critical requirements included high reliability, accuracy, and low synchronization times. The system had to avoid adversely affecting delivery drivers' performance while achieving its cost savings. The previous system sometimes could not retrieve delivery receipt images because the error rate associated with the paper tracking system was so high. The opportunity for cost savings was millions of dollars per year in lost revenue when customers demanded redelivery for goods they claimed they did not receive.

Avoiding these costs justified developing an improved system based on palmtop devices with integrated bar code scanners, client applications for tracking, and digital signatures. However, more accurate delivery provided the real savings, which not only helped the bottom line but also enhanced customer satisfaction.

The existing hardcopy-based system was labor-intensive and error-prone. Drivers received hardcopy delivery schedules, manifests, and receipts for customer signatures. When they returned, clerks scanned these paper documents into an imaging system for later use as proof of delivery. Being paper, schedules and receipts could get lost, making it difficult to evaluate driver performance and forcing redelivery when customer inventory systems did not reflect deliveries. The paper-based system also did not allow for positive validation that a particular container had been delivered to the correct customer, sometimes resulting in incorrect deliveries.

The new pervasive system manages deliveries across the North American continent by over 60 courier companies. McKesson has over 40,000 customers, 30 distribution centers, and 2,000 couriers. The courier companies had a wide variety of expertise levels in information technology, from those needing a complete turnkey solution, to those having their own advanced database and inventory systems. The average system load is about 20 concurrent synchronizations, totaling roughly 80,000 transactions per day. This environment required simplicity, robustness, and cost-effectiveness—and it had to be operated by a technically unsophisticated field force.

When Bork began the system design four years ago, he immediately recognized a need for communication and interface tool layers, which would require significant investment to develop. Developing them for his project alone clearly would not be cost-effective, so he investigated offerings from outside companies. At that point, there were no comprehensive solutions available, so he began to work with AvantGo, a startup then with only six or seven employees but with a middleware product plan that seemed well suited to his project.

During the initial phases, as the middleware tools evolved, the Closed Loop Delivery System's architecture evolved in parallel to take advantage of improv-

PERVASIVE MOBILITY TOOLS FROM AETHER SYSTEMS

In addition to AvantGo, Aether Systems (www.aethersystems.com) offers another commercial middleware toolkit for mobile workers using wireless PDAs. Called Fusion, Aether's product is a layered platform for secure, reliable extension of enterprise databases to heterogeneous wireless mobile devices. Sun Microsystems deployed it for field force automation, and Staples office supplies uses it for delivery tracking. Both are large-scale mobile applications, similar to AvantGo's application at McKesson.

Fusion comprises layers between mobile-wireless client devices and enterprise databases and applications. Each fills a specific role in enabling applications to function properly on a range of target devices and can be tailored to specific applications, services, network protocols, and devices. This way, users can assemble a complete stack for specific requirements. Each Fusion layer knows its own role and interfaces above and below, which lets Fusion components interface with other vendors' technologies and each other in the stack. The component layers include client applications; network adapters; communications adapters; core services (such as authentication, encryption, compression, billing, network and user management, and device discovery); service adapters (such as synchronization, streaming media, voice communications, and transcoding); and data adapters (interfaces to standard applications and APIs such as Exchange, Notes, ODBC, XML, and HTML).

Additionally, Aether Intelligent Messaging tools perform key functions in three areas:

- Secure wireless transport—using standard cryptography algorithms and procedures
- Development support—offering an environment that reduces the complexity of extending applications to handheld devices over multiple wireless networks
- Compression—reducing the number of data packets in a typical transmission, so information can travel faster and at lower cost, for bandwidth savings.

Aether Fusion is now available on Sharp Zaurus, Symbol SPT 1700, RIM (described in the *IEEE Pervasive Computing*'s April–June issue), and other wireless PDAs. It supports email with attachments, wireless Web browsing, and application-specific communication.

ing capabilities. Because he was an early adopter, Bork's requirements helped define the incremental improvements to the toolset, but this required adapting to multiple builds as the middleware layer was enhanced.

Bork cautions that deploying a system to replace a multibillion-dollar enterprise's core business processes requires

- A careful and systematic implementation plan with prototype rollouts as a proof of concept
- Substantial field trials
- Effective end-user training
- 24-hour help desk support
- An owning organization for servers and infrastructure
- A development organization to respond to error reports and engineering change requests

Success requires early buy-in from diverse constituent groups. In addition to providing the routing instructions, inventory, and container lists for drivers, remote software configuration management for the palmtop devices is also crucial.

Major system components used included the Symbol SPT 1700 and 1800 rugged Palm OS terminals with barcode reader, a signature-delivery application, the AvantGo client layer, and a back-end AvantGo Enterprise Server component (see the "Symbol Technology's Rugged Pervasive Devices" sidebar). Bork migrated this server to Sun Solaris systems as the application scaled up into production. Between the clients and servers, the AvantGo Mobile Link provides for synchronization, operating in batch mode, which lets drivers synchronize with the enterprise Oracle database before departing the distribution centers in the morning and returning in the evening.

The AvantGo virtual server component, running on the palmtop client, lets drivers process HTML forms during the day, with the virtual server handling the upload and synchronization when they return. Flash memory is reserved for data updates at each customer site, so the system can retrieve delivery confirmations when the device returns to its home cradle, even if the battery runs out during the route.

The courier organization's financial and physical constraints drove hardware selection, as did the parent organization's goals. The smaller distribution centers might consist of a simple transshipment dock with a small office and a single shared telephone line. Larger ones might have multidock warehouses, extensive communications infrastructure, and their own inventory control systems. Cost was also a strong consideration, because thousands of palmtops would be deployed. Battery life was critical, so they chose a small, efficient operating system and preferred small-footprint tools requiring less memory.

The target specification was 2 Mbytes of RAM and 2 Mbytes of flash memory but soon grew to 8 Mbytes of RAM and 4 Mbytes of flash memory as the Sym-

SYMBOL TECHNOLOGY'S RUGGED PERVASIVE DEVICES

As pervasive computing transitions from a research field to a commercial reality, many companies are entering the marketplace. Palm, Microsoft, the Linux community, and others offer pervasive operating systems. There are many handheld devices such as those from Palm, HP/Compaq, and Sharp. There are also comprehensive offerings for industrial users, such as the Symbol 1700 series devices discussed in the main story.

In taking a closer look at Symbol's product line, you'll find a variety of industrial pervasive devices, including notepad palmtops, pen-based tablets, handheld and wearable scanners, vehicle-mounted terminals, and mobile printers (see Figure A).

Figure A. Symbol's product line: (1) Symbol SPT 1800—palmtop with barcode scanner and signature capture; (2) Symbol PPT 2800—for any unruly Klingons you may encounter; (3) Symbol SRS-1—wearable ring scanner; and (4) Symbol VRC 7900—vehicle-mounted terminal.

bol devices matured. The flash memory was loaded with a custom OS build and AvantGo services, with the remainder reserved for confirmations accumulated during the deliveries. This configuration permitted a battery life of several working days. The system required different sync cradles and networking configurations, including standalone corporate PCs, Ethernet, and dial-up point-to-point protocol supported by an ISP-like connection.

THE MCKESSON CLOSED-LOOP DELIVERY SYSTEM PERVASIVENESS REPORT

Because this issue's department has a strong focus on a single application, the McKesson Closed Loop Delivery System, the Pervasiveness Report follows. By nature, pervasive systems are closely integrated to both human activities and conventional systems infrastructure, and Table A shows a short take on how this application is positioned. Systems with an average pervasiveness index of medium or greater are significantly pervasive. A high rating for a category means that the function is broadly diffused across, or central to the application, while a low rating means that either the function is not present, or it is not fully integrated into the application.

Table A. Pervasiveness report for McKesson Closed-Loop Delivery System

Category	Function	Rating
Computing infrastructure	Pervasive middleware, Web servers	High
Mobile computing devices	Palm OS Symbol terminals with barcode scanner for data collection	High
Wireless networking	IR and Optional RF depending on configuration	High
Device discovery	Preconfigured ISP-like functionality	Low
Service discovery	Static server configuration	Low
Multimodal user interfaces	Pen, digital ink, keys	Medium
Pervasive databases	Pervasive middleware-mediated access to enterprise databases	High
User authentication	Statically configured	Low
Secure networking	Encrypted VPN provided	High
Task integration	Essential to task	High

By design, a single support call would let the user set up the sync cradle with the appropriate dial-up and Internet protocol information. Although the Symbol SPT 1700 devices were not ready at the development cycle's onset, they subsequently performed well, with very few returning for broken screens or other maintenance failures despite the physically difficult environment of the loading docks and the routes themselves.

Bork considered wireless connectivity for en-route processing—and it might yet be added—but it is not widely enough available yet, even though there is a rugged palmtop device with wide-area network capability. Because the AvantGo virtual server component lets the Web-based interface operate through outages, continuous connectivity was not actually critical to the system as architected. However, as wireless coverage becomes more prevalent, it can be added without modification to the existing application suite. That's because AdvantGo developed it using APIs from a middleware layer that can either use a communication link or a local cache pending uplink availability.

The application has now been deployed across about 80 percent of McKesson's network. In contrast to the previous-generation paper-tracking system, the drivers now synchronize their palmtop devices with up-to-date delivery and routing instructions. At the customer sites, drivers scan product containers as they are delivered, setting off an immediate error indicator if the wrong containers or an incomplete set of containers is scanned. If the delivery is correct, the system captures the customer signature electronically, adding it to the HTML transaction forms for subsequent update.

Inventory accuracy is now much higher and delivery errors have substantially been eliminated, reducing claims for incorrect deliveries.

Based on pervasive devices and pervasive middleware, the new system has dramatically improved McKesson's quality of service. Bork reports that inventory accuracy is now much higher and delivery errors have substantially been eliminated, reducing claims for incorrect deliveries. The forms imaging system, with its high labor cost and error rates, has also disappeared. Future enhancements might include continuous connectivity for more immediate status tracking and dissemination of the system into the delivery network's remaining 20 percent. For now, though, the initial system release has proven highly successful.

Vince Stanford is the lead engineer for the NIST Smart Space Laboratory, project manager for the NIST Smart Space project, and a founding member of *IEEE Pervasive Computing* magazine. Contact him at vince-stanford@users.sourceforge.net.