IEEE distributed systems ONLINE
Expert-authored articles and resources

IEEE DISTRIBUTED SYSTEMS ONLINE 1541-4922 © 2006 Published by the IEEE Computer Society
Vol. 7, No. 3; March 2006

Spotlight: The Rise of the Smart Phone

Pei Zheng, *Microsoft*
Lionel M. Ni, *Hong Kong University of Science and Technology*

Cell phones are everywhere, growing in both number and type. Although we lack a clear distinction between a cell phone and a *smart phone*, generally, a smart phone is a next-generation, multifunctional cell phone that provides voice communication and text-messaging capabilities and facilitates data processing as well as enhanced wireless connectivity. You might consider the smart phone as the marriage between a powerful cell phone and a wireless-enabled PDA.

According a study by UK market research firm Canalys (http://www.canalys.com/pr/2005/r2005102.htm), more than 12 million data-ready wireless mobile devices shipped in the second quarter of 2005—twice the fewer than 6 million shipped in the year before. Although smart phones account for only a small slice of the worldwide cell phone market, many consumers are opting for business-centric smart phones as the price continues to drop.

We're riding a so-called "fifth wave of computing"[1]—the rise of wireless-enabled computing devices empowered by ubiquitous, low-cost wireless data connections and open software standards. In other words, we have a growing need for personal computing and on-the-go communication supplemented by the rollout of 3G (third generation) cellular services and widespread Wi-Fi hotspots. This is motivating mobile software platform providers, independent software vendors, and cell phone manufacturers to team up and offer applications and services targeting common consumers and enterprises, using smart phones as a universal end point for data access and networked computing. Already, smart phones offer the functionalities of a music player or a digital camera. Additionally, some have a built-in smart card for payment and ticketing, and doctors, salespeople, the police, and so on use them for data collection and real-time data access.

The big picture

Unlike most conventional cell phones, a smart phone will have these features:

- A color LCD screen with backlight.
- Enhanced wireless capability such as Wi-Fi, Bluetooth, and infrared and the ability to synchronize with computers.
- A large memory (RAM and ROM) and persistent storage (memory cards or built-in hard disk).
- An advanced operating system with a set of applications that usually include games and calendar, scheduler, address book, media player, book reader, recorder, note, and calculator functions. Many have a camera; some even have a Carl Zeiss lens.

Additionally, smart phones generally fall into three categories in terms of handset design, representing three camps in the industry sector:

- high-end cell phones by cell phone manufacturers, such as Nokia, Ericsson, and Motorola;
- PDA phones by HP and Palm; and
- enhanced wireless email devices (that is, Blackberry) by Research in Motion.

However, software platforms for smart phones aren't aligned with this categorization at all.

Cell phone manufacturers used to develop their own proprietary, highly customized operating systems for their product lines. Because independent software vendors generally don't have access to these operating systems, they're not suitable for today's vertical wireless market. On the battlefield of smart phone software platforms, we have only a few big combatants: Symbian OS, Microsoft Windows Mobile, Palm OS, and some variations of embedded Linux systems.

Symbian OS is clearly the market leader, powering more than 32 million cell phones and smart phones by Nokia, Sony Ericsson, Motorola, Samsung, and others. Several smart phones—such as HP iPaQ Pocket PC phones and some Motorola smart phones—use Windows Mobile. And until recently, Palm OS was the only operating system for Palm PDAs and Palm smart phones. On the other hand, Linux is gaining some traction in this market as well. According to Trolltech, the company that developed two popular software development tools Qt and Qtopia, more than 20 cell phone manufacturers are building Linux phones.

These operating systems are highly optimized for resource-constrained cell phones and smart phones and are becoming very sophisticated in supporting multithreading, advanced power management, and a few types of wireless capabilities. On the application development side, every software platform vendor is actively promoting its platforms and software development kits (SDKs) by offering incentive marketing programs.

For an overview of smart phone and wireless-technology trends, we recommend chapter 2 of our book *Smart Phone and Next Generation Mobile Computing*.[2]

The handset

A smart phone's hardware components usually include a microprocessor, a mainboard, an antenna, ROM, RAM, a battery, additional storage such as flash memory or an SD (secure digital) card, network interfaces, and an LCD or a TFT (thin-film transistor) screen. Some smart phones have a tiny hard disk.

Processor

Processors for smart phones include ARM (advanced RISC machine), Motorola Dragon Ball, MIPS, and TI OMAP. Intel licensed ARM technology and uses it in its XScale lines. The XScale architecture leverages Dynamic Voltage Management technology, which lets the processor's operating voltage and frequency scale dynamically in response to varying computing and communication needs. Some Palm PDAs and smart phones use Motorola 68000-based Dragon Ball processors. Some game consoles and cell phones are using MIPS and TI OMAP.

Usually these processors run at several hundred megahertz. Intel's researchers have demonstrated 1-GHz mobile processors. In addition to increasingly high speeds, another trend is the use of SoC (system on a chip) technology, which lets a processor incorporate a set of distinct functionalities in the same package.

Battery

Researchers have been working to address the important issue of smart phone power efficiency. Smart phones typically use three types of cell phone batteries: NiMH (nickel metal hydride), Li-ion (lithium-ion), and Li-polymer—all of which support a few hours of talk time and a week of standby time. Wireless communication, such as Wi-Fi, generally consumes power more quickly than simple computing tasks. Fuel cell batteries and other new battery technologies for mobile devices have emerged but aren't in mass production.

Memory and storage

Smart phone operating systems and applications are comparatively smaller than their desktop counterparts. So, it's possible and highly desirable to put all system and application code into RAM, ROM, or flash memory. Many smart phones have 64 to 128 MBytes of SRAM for application code, 128 to 256 MBytes of flash memory for system code, and more than 128 MBytes of flash memory for user data. People also use flash memory for external removable data storage—for example, SmartMedia, CF (compact flash), MMCs (multimedia memory cards), and SD (each of these refers to an industry standard supported by a number of companies). CF memory cards can supply as much as 4 GBytes of storage capacity. Then again, some smart phones are starting to offer Gigabit hard disks.

Screen display

Smart phone screen displays are designed to facilitate various applications, including Web surfing, email, and audio and video playback. Screen sizes range from 2.2 inches (5.588 cm) to 10 inches (25.4 cm). Screen resolution continues to improve—some smart phones have QVGA (320 × 240) or VGA (640 × 480) displays with more than 64,000 colors. At Philips and Lucent Bell Labs, researchers are developing a flexible display that users can roll up and even fold, which could lead to some interesting mobile device displays.

Keyboard or keypad

Many smart phones adopt the traditional 12-button cell phone keypad, plus navigation and function keys. Such a layout allows for one-hand operation. Some smart phones use a QWERTY layout, a tiny version of the standard computer keyboard. Because all letters are lined up, the keyboard is larger than a phone keypad and the keys are quite small, making typing somewhat difficult. Some other smart phones, especially PDA phones, provide a stylus for handwriting input. The application UI design must take into account the keyboard or keypad type and try to minimize text input.

Software platforms

A clear trend for software platforms has emerged: feature-rich, extensible operating systems that support third-party applications are replacing cell phone manufacturers' proprietary systems. Like the desktop operating system market, only a few major players exist.

Each of these software platforms provides a set of APIs and programming tools. Additionally, to ease testing and debugging on a real smart phone, these platforms' SDKs usually come with a software emulator that lets developers deploy and test code without a physical device.

Symbian OS

Symbian OS is developed by Symbian, a company supported by several large cell phone manufacturers, including Nokia, Ericsson, Sony Ericsson, and Samsung. Originally based on the EPOC operating system, Symbian OS defines several UI reference models for different types of devices—for example, the Crystal (for regular cell phones) and Serial 60 (for smart phones) models. Symbian OS is a real-time, multithreaded preemptive kernel (versions before 8.0 don't have a real-time kernel) that performs memory management, process and thread scheduling, interprocess communication, process-relative and thread-relative resource management, hardware abstraction, and error handling. Its key features include mobile telephony supporting WCDMA (wideband code-division multiple access), GSM (global system for mobile) and GPRS (general packet radio service), and CDMA (code division multiple access) 2000 1x-RTT. Symbian OS also includes messaging services supporting SMS (short message service), multimedia, and common transmission control protocols. The latest version of Symbian OS is 9.

Symbian OS uses EPOC C++, a pure object-oriented language, as the supporting programming language for both system services implementations and APIs. It also allows Java applications for mobile devices (Java 2 Micro Edition, J2ME, applications) to run on top of a small Java runtime environment. Symbian OS implements a CLDC/ MIDP (Connected Limited Device Configuration/Mobile Information Device Profile) 2.0 profile of J2ME specifications. The Web site (http://www.symbian.com/developer) provides numerous technical documents of Symbian OS, SDKs, and sample code, as well as information on Symbian OS development and the Symbian developer community.

Windows Mobile & the .Net Compact Framework

Microsoft Windows Mobile (http://www.microsoft.com/windowsmobile) consists of a Windows CE-based operating system, a runtime and programming framework (the .Net Compact Framework), and supporting software development tools. The .Net CF provides runtime support for managed applications that use a set of .Net CF classes. The .Net runtime also handles memory management and type safety. The .Net CF consists of a CLR (Common Language Runtime) specifically designed for resource-constrained mobile devices and the .Net CF class libraries. This CLR is much smaller and more efficient compared to the desktop CLR. Windows Mobile has two flavors: Windows Mobile for Pocket PC and Windows Mobile for Smart Phone. Microsoft has recently released Windows Mobile 5.0 for Smart Phone.

Mobile software development with .Net CF (http://msdn.microsoft.com/smartclient/ understanding/netcf) is enabled by Microsoft's programming tool, Visual Studio .Net (the latest version is VS .Net 2005), which supports smart-device projects including the Pocket PC, Smartphone, and Windows CE projects. Developers can choose either C# or Visual Basic as the programming language. As part of the Smartphone platform offering, Microsoft also provides a Smartphone SDK for C and C++ unmanaged application programming using Win32 (WinCE) APIs, bypassing the .Net CF runtime.

Palm OS

Before Palm's legendary Palm Pilot PDA, the PDA market barely existed, although Apple and some other companies did have some pioneering products. For a long time, Palm PDAs running Palm OS (http://www.palmsource.com/palmos) have been dominating the PDA market. Palm OS is a preemptive, multitasking operating system that provides protective memory management and HotSync support. With version 6, Palm OS, namely Palm OS Cobalt, started offering built-in Bluetooth and Wi-Fi support. Palm OS supports both the ARM and Motorola 68000 architectures.

Developers can choose programming languages from C, C++, Visual Basic, or Java, although C is most widely used for Palm OS software development (http://www.palmsource.com/developers). Palm OS application development is facilitated by the Palm OS 68K and Protein SDKs and some commercial *developer suites*. A developer suite is an integrated software tool that lets developers create both ARM-native and Palm OS Protein-powered applications for Palm OS Cobalt and 68K applications. Commonly used developer suites are Metrowerks CodeWarrior (http://www.codewarrior.com), PRC-Tools (http://www.palmos.com/dev/tools/gcc), or Eclipse (http://www.eclipse.org) for a cross-platform programming environment.

Embedded Linux

Linux is a major player in the embedded-systems market—many networking devices and consumer electronics devices run a variant of Linux. (See the *Linux Journal*'s tutorial on embedded Linux systems, http://www.linuxjournal.com/article/4322). In the case of smart phones and cell phones, some manufacturers such as Motorola and NEC chose embedded Linux, a highly customized variant of Linux, as the underlying operating system for some of their products. Embedded Linux supports multithreading and real-time application development.

Application development on open source or commercial embedded Linux systems differs from system to system. Aside from vendor-specific tools, developers often use standard Linux toolchains with C/C++ libraries. For example, Qt/Embedded (http://www.trolltech.com/products/embedded) is a popular GUI development framework optimized for Linux-powered mobile devices. Widely used embedded application development tools include GnuPro Toolkit (http://www.redhat.com/software/gnupro) and the well-known Gnu toolkit.

Java and BREW

Java and BREW (Binary Runtime Environment, http://brew.qualcomm.com/brew/en) are both middleware technologies for smart phone computing. Java's J2ME defines two software models, or configurations: the Connected Limited Device Configuration (CLDC) for mobile devices and the Connected Device Configuration (CDC) for consumer electronic devices.

The Sun Java Wireless Toolkit (http://java.sun.com/products/sjwtoolkit/index.html) is a toolbox for developing J2ME CLDC configuration applications. You can download both CLDC and CDC from the Sun J2ME Web site (http://java.sun.com/j2me). Accommodating mobile resource-constrained devices requires using an optimized virtual machine called the kilobyte virtual machine. KVMs exist for Symbian OS, Palm OS, Embedded Linux, Windows Mobile, and BREW, which is a proprietary software platform developed by Qualcomm.

Courses on mobile application development

The proliferation of various mobile applications calls for more mobile developers who can use those tools to write programs targeting a variety of platforms. Aside from numerous training programs provided by software vendors, some universities (such as Indiana University (http://www.cs.indiana.edu/~connelly/b490_Spring05.htm), the University of Southern California (http://www.usc.edu/dept/publications/cat2004/schools/engineering/information_technology.html Mobile Game Programming), and Washington University in St. Louis, http://www.cait.wustl.edu/courses/WDE200.co) are offering courses on mobile-application development.[3] Companies such as Microsoft have also offered support for such courses (including senior capstone projects) in the form of mobile devices and software development tools.

Services and applications

We choose three categories of emerging services and applications illustrate how smart phones can provide a richer user experience and how these applications and services could shape existing enterprise computing and business processes. Some wireless service providers offer other applications and services such as mobile video and music downloads. Chapter 7 of our book discusses in detail next-generation smart phone services and applications.[2]

Location-based services

A new set of wireless services uses positioning techniques to provide personalized, location-based information about cell phone users. Cellular technologies can inherently find and track a cell phone using signal triangulation from base stations. In fact, many of today's cell phones are location-capable owing to US E911 and Europe E112 regulations that let you locate (to within 50 to 100 meters) a phone user who placed an emergency call. GPS is probably the best outdoor location technology in terms of precision and cost; some cell phones have a built-in GPS receiver. But GPS requires line of sight and doesn't work well in "urban canyons" (that is, close to a high building or under a tree) where the GPS receiver can't communicate with the satellite. Moreover, a GPS receiver normally takes a while to fix its position. With assisted GPS, base stations equipped with GPS receivers communicate with a powerful assistance server that computes cell phone locations.

Wi-Fi (IEEE 802.11) can be used as another major wireless location-sensing technology. One way this works is to build a system that generates a wireless location fingerprint with signal samples from multiple access points. For example, Intel Research's Place Lab (http://www.placelab.org) project[4] aims to provide a low-cost, easy-to-use device-positioning platform for location-based applications. The Place Lab software lets a mobile device (such as a PDA or a cell phone) derive its location from radio beacons collected from wireless LAN access points, GSM cellular base stations, and fixed Bluetooth devices in the area, whose positions are available in a location database that has location data for as many as 2.2 million radio beacons. The location database is built based on wardriving data repositories, the FCC's cellular base station database, and the location data of wireless LAN access points from companies, universities, and so on.

Another important element of location-based applications is the map—not the traditional road map but what we call a *geomap,* which incorporates geographic information, roads and highways, business information, demographic information, and anything else that you can associate with a specific location. Most location-based application users aren't likely interested in raw geographic data; rather, they might want to know where to find the closest gas station or whether a shopping mall has a sales promotion going on. Consequently, it's highly critical for maps to supply users with sufficient data for results with good precision. Navtaq and similar companies supply service providers (such as MapQuest and Google maps) with this kind of data.

In addition to traditional maps and navigation assistance services, smart phones have some other interesting location-based applications:

- *Mobile local search.* MLS users might want to find information about nearby, specific businesses or friends. Some services combine this with wireless advertising, which sends electronic coupons and special offers only to customers in the vicinity. Google, Yahoo, and Microsoft MSN all provide mobile local search, and numerous companies—such as go2 (http://www.go2.com) and m-spatial (http://www.m-spatial.com) —are developing services that can offer more than just a Web search result displayed on your phone screen.

- *Mobile social networking.* Using cell phones and smart phones for social networking has gained significant traction. Online services such as Dodgeball, http://www.dodgeball.com, (part of Google now) let users send text messages of their whereabouts and pictures to friends and friends' friends who are within a few blocks.

- *Location-enhanced asset management.* Warehouses, retail stores, and manufacturing plants can leverage indoor location-sensing technologies using Wi-Fi, Bluetooth, or RFID (radio frequency identification) to improve efficiency and productivity of inventory tracking and supply chain management. Ekahau (http://www.ekahau.com) and Newbury (http://www.newburynetworks.com) are examples of such systems.

To build location-based applications, developers might use location-based software solutions that bundle database access and programming interfaces. For example, Microsoft MapPoint (http://www.microsoft.com/mappoint) can provide maps, navigation assistance, proximity searches, and location-related business intelligence as programmable XML Web services. A MapPoint system interfaces to a wireless service provider's location service and integrates it with data such as business logic information and POIs (points of interest) for its mobile users. For an enterprise's specific business needs, a MapPoint location server acts as the central control linking mobile users, MapPoint Web services, and location services from wireless services providers.

M-commerce

Using a cell phone with built-in infrared or Bluetooth to pay at a vending machine or a convenience store is nothing new—Japan, Korea, and some European countries have had this kind of m-payment service for a few years. In fact, m-commerce encompasses a much broader vision of mobile services and applications, including shopping and payment, banking, and ticketing. Many believe that we can clone the tremendous success of online retail services in the mobile world. But realizing this requires solving numerous challenges.

First, the smart phone's form factor makes it difficult for users to navigate through common Web pages on a phone display, thus impeding the acceptance of many Web based e-commerce applications and services. Rather than considering porting legacy Web-based e-commerce applications to the smart phone, we might need to focus on those services and applications that exclusively target people on the move and are designed to run on smart phones.

Second, using multiple wireless technologies in m-commerce raises several mobile-security issues. We're well aware of the vulnerabilities in Wi-Fi's WEP (Wired Equivalent Privacy) mechanism and Bluetooth's default wide-open security mode that enables attacks such as bluesnarfing and bluebugging.[5] Moreover, viruses and worms have invaded the mobile world; not to mention the tens of thousand of PDAs and cell phones lost every year, putting sensitive personal or business data at risk. A smart phone isn't just a voice telephony device any more—it carries, processes, and sends and receives sensitive data such as confidential business information and financial data. So, both the mobile terminal (smart phone) and the wireless infrastructure must be highly secured against any device- or network-centric attacks.

Third, wireless links lack the stability of wired network connections, so designers of mobile applications often use techniques such as *aggressive caching*—offline processing in disconnected mode. This inevitably complicates the design of m-commerce systems.

These issues call for research and development in mobile user interface design, mobile middleware, and mobile biometrics, and, on the network infrastructure, convergence of computing and communication and integration of heterogeneous wireless networks. For example, to secure access to a smart phone or a PDA, we could apply lightweight biometric techniques such as fingerprint scanners. Additionally, to address serious service interoperability issues, numerous industry consortia have been formed—such as Mobey Forum (http://mobeyforum.org) and the European Committee for Banking Standards—to promote standardization of m-payment systems and m-banking systems.

Mobile enterprise applications

Enterprise applications such as ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), and SCM (Supply Chain Management) might use a smart phone as an always-on end point to provide real data access and transaction support for employees at a customer's site or on the road, for doctors visiting a patient, or for police officers on the street. For example, an insurance agent at a traffic accident site could use an enterprise application portal, operating on top of cellular connections available on a smart phone, to quickly assess the claim and file a report with pictures. You can enhance SCM systems to support real-time pallet and goods tracking with RFID technology. Both indoor and outdoor location-based technologies play an important role in mobile-enterprise arena, as they're the foundation for next-generation enterprise applications that value location information for business processes.

In addition to augmenting traditional enterprise applications, companies might consider using the wireless data transport provided by the underlying wireless infrastructure to improve collaborations between employees on the move or in branch offices. Other than placing voice calls, field engineers, salespersons, and delivery personnel can use smart phones to send instant messages, pictures, and even short video clips to each other or to a group. You could use their location, presence status (as shown in an enterprise instant-messaging application), and work progress for dynamic work assignment tracking and scheduling. These enhanced collaboration methods form an integral part of the enterprise system, allowing employees on a remote site to access these enterprise applications from a mobile portal running on a smart phone via cellular data or other wireless connectivity. So, they're much more effective than regular voice calls in improving work productivity and reducing operational cost.

Conclusion

Thanks to the rapid advancement of wireless technologies, a whole new set of mobile wireless services and applications are set to empower people by providing communication and computing anywhere, anytime. Moreover, smart phones—the next-generation multifunctional, multiwireless cell phones—will likely be the mechanisms that deliver these services and applications. Both the industry and academic research communities are working to resolve numerous issues, ranging from low-power hardware design, to flexible I/O methods, to location-sensing technologies, to mobile security issues. Nevertheless, with the deployment of 3G and Wi-Fi hotspots nationwide and the adoption of smart phones by the masses, we expect to see a spectrum of new services and applications targeting smart phone users that essentially mobilize people's daily lives and enterprise computing.

References

1. 1. M.V. Copeland , "How to Ride the Fifth Wave,"*Business 2.0,*July 2005; http://money.cnn.com/magazines/business2/business2_archive/2005/07/01/8265500.
2. 2. P. Zheng and L.M. Ni , *Smart Phone and Next Generation Mobile Computing,* Morgan Kaufmann, 2005.
3. 3. H. Artail , "A Multiplatform Methodology: Developing Mobile Device Applications," http://doi.ieeecomputersociety.org/10.1109/MPRV.2005.24, *IEEE Pervasive Computing,* vol. 4, no. 2, 2005,pp. 92-96.
4. 4. A. LaMarca , et al., "Place Lab: Device Positioning Using Radio Beacons in the Wild,"*Proc. Pervasive 2005,* LNCS 3468, Springer, 2005, pp. 116-133.
5. 5. D. Dagon , T. Martin and T. Starner , "Mobile Phones as Computing Devices: The Viruses are Coming!" http://doi.ieeecomputersociety.org/10.1109/MPRV.2004.21, *IEEE Pervasive Computing,* vol. 3, no. 4, 2004, pp. 11-15.

Pei Zheng is a software engineer at Microsoft. His research interests include distributed systems, network simulation and emulation, and mobile computing. He received his PhD in computer science from Michigan State University. He is a member of IEEE. Contact him at One Microsoft Way, Redmond, WA 98502; peizheng@microsoft.com.

Lionel M. Ni is the Chair Professor and Head of the Computer Science Department at the Hong Kong University of Science and Technology. His research interests include high-performance computer systems, high-speed networks, pervasive computing, wireless sensor networks, and distributed systems. He received his PhD in electrical and computer engineering from Purdue University. He is a fellow of IEEE. Contact him at the Dept. of Computer Science, Hong Kong Univ. of Science and Technology, Clear Water Bay, Kowloon, Hong Kong; ni@cs.ust.hk.

Cite this article: Pei Zheng and Lionel M. Ni, "The Rise of the Smart Phone," *IEEE Distributed Systems Online*, vol. 7, no. 3, 2006, art. no. 0603-o3003.