Part II

Wireless Internet Applications

A *wireless application* is software that runs on a wireless device that exchanges content over a wireless network. The actual wireless applications are distinguished from one another based on the wireless devices, networks, and application families, which we covered in Part I and summarize in the following list.

Devices
- Web phone
- Handheld
- Pager
- Voice portal
- Web PC
- Communicating appliances

Networks
- WAN
- LAN
- PAN

Applications
- Messaging
- Web browsing
- Interacting
- Conversing

It is time to look more closely at wireless applications that perform messaging, Web browsing, and other interactive and conversational functions. To understand which devices work best, which networks can deliver good service for what mobile audiences want to do, and the extra technology that can be applied, it is helpful to see real applications, to consider the thought behind their development, and to learn how they are made.

There is a nearly universal process to develop wireless applications that run on the wireless Internet. The wireless development method we show in Part II works for a 3G Web phone microbrowser over a W-CDMA network, a handheld interactive

business application over a wireless Ethernet LAN, an interactive Java phone MIDlet over a WAN, a pager over a Motient network, or even a voice portal using AMPS.

Many developers first think that their wireless cell phone applications have to connect to some entirely foreign back-end modem bank they have to learn about, or that they have to program for some special communications server. That is not the case at all. Wireless applications are generally fielded from the same Web server that PC Web browsers read from. All content originates from the Internet. A Web phone microbrowser fetches Internet Web content that is generated from a Web server. It is the server that supplies the content stream to the wireless application. Most of the development for the Web phone page design is actually done on the Web server.

Wireless applications are different from PC applications in that the guts of the content are expected to reside elsewhere. Standalone PC applications like spreadsheets continuously operate on local storage, whereas wireless applications operate temporarily and connect to exchange their underlying data at a moment's notice. Personal mobile devices often include personal information management (PIM) applications such as address books, contact lists, and calendars. PIM applications synchronize with a PC using a cable. The PC acts as a personal hub to send to and receive from other parts of the Internet. Synchronization can occur over the air with a PAN or LAN, although WAN synchronization can be expensive. Distributing and coordinating data, such as a group calendar, is an emerging form of wireless application. The content may be coordinated with a master calendar server or simply be distributed, perhaps using SyncML.

To learn the process of wireless development, it is helpful to understand the business case. Look for this in chapter 16, in the section "The Site: Wireless Business Models," where we cover the business reasons for building the application. Assuming there is a business justification or motivation, then the wireless device, network, and application must be selected and developed. In chapter 6, "Concepts for Working with Wireless Applications," we show the universal project development process used for an Olympic Event WAN browsing application, a WAN handheld travel guide, a LAN interactive building inspector application, a PAN device application, an SMS messaging application, and a voice portal horoscope application. Each project involves four steps:

1. *Introduce the person.* Start wireless projects by identifying the mobile user. Successful developers take time to characterize a persona, introduce themselves to mobile users, and become familiar with their patterns of work.
2. *Build the content.* Build a database with real content first. Real content that appears early in a project will drive many later decisions. Content is the stuff that users gather, send, and receive. The detailed form of content, such as text or geocodes, should relate to the user persona study.

3. *Develop the application.* The logic of the way the content is or will be used comes next. This involves techniques for creation, access, navigation, and ordering of the application and its content. This stage typically uses simulators.
4. *Use a real wireless network and device.* Exactly how the content is styled and how the interface is physically controlled are refined at the end of the development process. At this point, exceptions are made for unique devices. The majority of text in this part covers implementation techniques for the unique devices of each network.

Often users think of applications by their real-time and offline qualities. WAN-based systems like Web phones require a constant real-time connection. LAN-based systems like handhelds are designed to work offline. Devices and networks give an application either a real-time or a delayed quality. Messaging applications may need real-time connections to be able to "chat." But most people use the mailboxes and answer when they can. Browsing the Web for real-time information obviously needs a connection that a passive offline browser does not require. Interactive applications on handhelds are designed to work exactly the same offline or when they are connected and can exchange data. In conversational applications, a direct call is common but voice mail can effectively accomplish the same thing. To the user, the many timely kinds of wireless connections are all various forms of personal connection.