6 Chapter

Concepts for Working with Wireless Applications

Our working knowledge of wireless Internet so far is a mix of concepts and trends. Trends such as "people will use more wireless devices than wired ones" help motivate and anticipate, but they do not really help us understand the principles to write software or deliver wireless content.

"To work for my wireless company, you should be familiar with these ideas," said the director. From this point forward we add the details of an application to the idea. There are six central ideas we have covered that you should know about and that we will illustrate.

1. *The person is mobile.* Seems obvious. This idea changes everything once you understand it. The wireless Internet keeps people in motion. Once a person is mobile, the wireless content becomes personal. The unique qualities of personal behavior are established in the section "Mobile Users Are the Secret" in chapter 2. The practice is discussed in "Defining Your Mobile Audience" in this chapter.
2. *Six devices mobilize the Internet.* Wireless devices need to be unified from a programmer's point of view. The characteristics of the Web phone, handheld, pager, voice portal, communicating appliances, and Web PC are introduced in "Close-up Characteristics of Wireless Devices" in chapter 9. A server solution

unifies them all, as we will see in chapter 18, "Building Servers and Matching Client Applications."

3. *Four wireless applications drive the wireless Internet.* Wireless application programming models include messaging, browsing, interacting, and conversing; they are compared in chapter 5. Programming examples of these applications appear throughout Part II.
4. *Three networks form the wireless Internet.* The three Internet networks are WAN, LAN, and PAN; they differ in power, regulation, and data rates. The key network advantage is the packet-switched network over the circuit-switched WAN network. Applications are grouped by networks in chapter 4.
5. *Mobile content defines the wireless business.* The value of content over software and hardware is detailed in chapter 5 in the section "Personal Content Drives the Wireless Internet." XML content defines portable data independent of style. Mobile content values of location, time, personalization, and transaction are covered in detail in chapters 7 and 8.
6. *Make location-based wireless applications and maximize mobility.* The mobile person wants relevant information. Developing location-centered applications gives them place and time, and can often be personalized. Examples appear in chapter 5 in the section "Showing Location" and in chapter 8, the section "Primary Location Applications."

Underlying terminology like *Mbps, MHz, 3G, 2G, 1G, CDMA, GSM, TDMA, AMPS* are introduced in chapter 4. The physics of the wireless Internet are covered in chapter 3 in the section "Wireless Spectrum."

Remembering That Small Is Beautiful

Software engineers coming from the world of a large, blazing, color screen, a rip-roaring 80 GB drive, 40 MB applications, and a two-foot long keyboard, look aghast at dim screens of little cell phones, pagers, and handhelds. Yet, how long would you last if you had to tote all that gear wherever you went? In motion all the time, you would appreciate lightness, smallness. Top mobile engineers design small systems, giving users what they want, at the moment, wherever they are. These new engineers live by a different code. Think small.

The U.S. auto industry provides an example of a move from large to small. In the 1950s the U.S. car industry had reached the zenith of large, stylized muscle cars that consumed maximum fuel and had wildly changing styles year after year. Under the hood, their engines needed frequent maintenance. Tires had to be replaced every year. The Europeans brought to market a totally counterculture

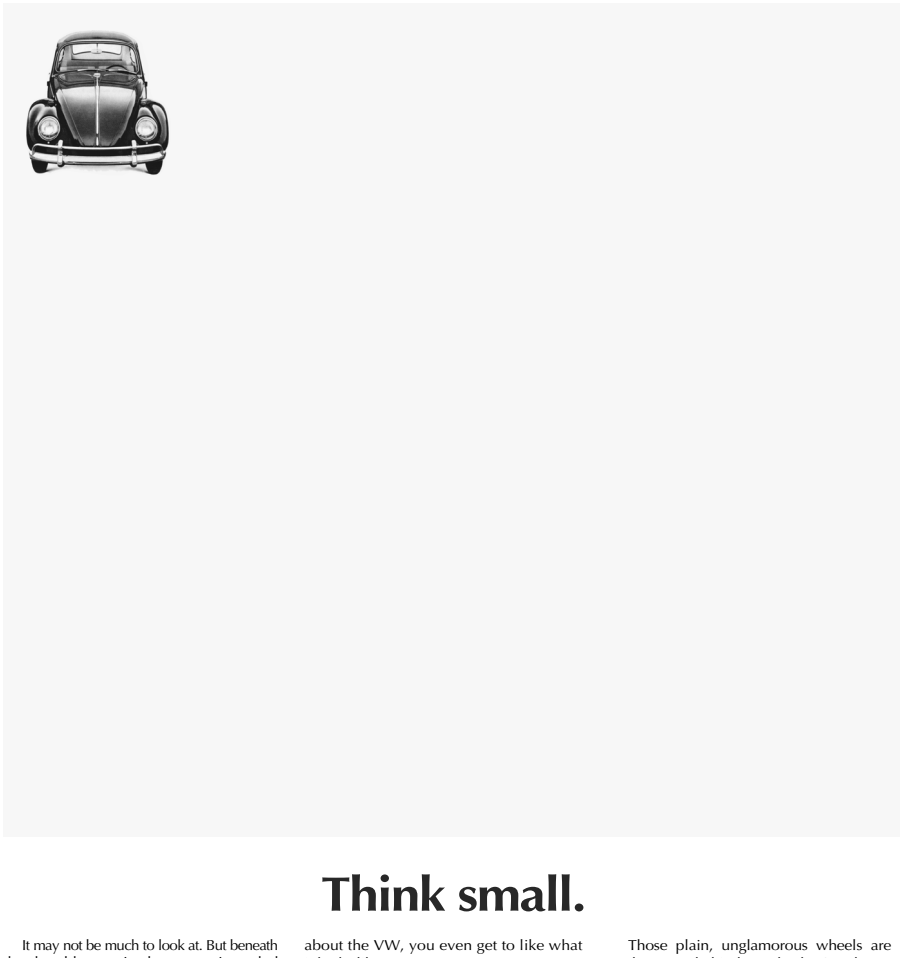


Figure 35 Examples of compelling mobile devices. The Europeans establish a useful consumer technology as the North Americans produce business systems. The Japanese are thinking smaller still. Pictured beside the VW ad are a Nokia phone, a Nextel HDML phone, and an NTT DoCoMo i-mode phone. (*Sources:* Advertisement copyright © Volkswagen of America, Inc.; Nokia phone copyright © 2001 Nokia Corporation. Nextel HDML phone copyright © 2001 Motorola, Inc. NTT DoCoMo, Inc. Printed with permission. All rights reserved.)

alternative vehicle, the VW Beetle, now revived in modern styling. The original VW was a shocking innovation. The technology changed automotive history and the honest understated advertising campaign changed marketing history. Rather than flashy and powerful, the German automobile was simple and adequate. The engine was the showpiece; it was gas efficient, air cooled, and mounted in the rear. What engineer could conceive of this? The homely VW was the original work of Ferdinand Porsche. His design used significantly fewer parts and simple systems that required lower maintenance than other automobiles. By eliminating the water pump, out went radiator failure. The suspension and tire system lasted an unheard-of 40,000 miles before a change.

Small hardware that runs all by itself is beautiful. Examples of compelling mobile devices (Figure 35) like the VW Bug have very small and efficient qualities. Unlike laptop PCs, the Web phone features a power-efficient operating system that runs for a much longer time before having to "fill-up" with a charge. It knows the time of day and knows where you are. It does not take three minutes to reboot. It is instant-on. Compared to the personal computer world, the wireless Internet is counterculture. It has its own content, its own culture. The vibrant mobile cultures do not care so much about "the Internet" as about utility and information. (Remember the i-mode story in chapter 1.) The counterculture inspires new ways of thinking about the possibility of communication and computation through the guise of mobile use.

When you think small, you make needless choices go away. When you work mobile, you do very specific bottom-line tasks. Good applications operate mostly as a wireless publishing channel. When you work offline and disconnected from the server, your system transparently queues messages and data. When the connection appears, it synchronizes updates, regardless of whether it is WAN, LAN, or PAN. At least that is how the professional systems work. Even system software is automatically updated over the air. Small wireless applications make use of one-click transactions. Ironically, to personalize wireless preferences for their small devices, users often work with their mega-gigabyte PC to set values and get content from large Web sites.

The relationship between the beautiful PC screen and the Spartan wireless small screen is like that between the muscle car and the VW Bug. A reminder of mobile utility (Figure 36) is that technology can be small and ugly, yet very reliable, taking you to useful places. Both the VW and its business model changed the standards of the world automotive industry. With the end of thinking in terms of "this year's model," it became necessary to produce only one repair manual and only one line of parts. This simplified manufacturing, inventorying, selling, and delivery to any place in the world. It was radical. It was simple. It was small.

Companies that took the trouble made a great leap forward with the Internet. The wireless Internet is another opportunity to eventually double the size of their

Figure 36 A reminder of mobile utility (*Source:* Copyright © Volkswagen of America, Inc. Printed with permission. All rights reserved.)

public. There is special work involved in building the small application, although a wireless application and a Web-based application "can share the same garage." Mobile and wired development have many common engineering properties. To think small, however, companies have to think hard about what functions their users most want and use; companies must take a fresh look at the mobile customer. When mobile people use a computer socially, their purposes are different from when they are desk-bound. Sometimes they enjoy new utility provided by GPS, Bluetooth, scanners, or expandable modules. But the vital beep of unglamorous paged text often brings smiles.

Today's Internet video commercials often excite us to ill-conceived impossible futures; meanwhile the mobile wireless applications in actual use are being produced for a revolution that is not televised. Utility often battles glamour. How could a simple technology like the 1959 Volkswagen succeed against the muscle of Detroit? Volkswagen, with an annual advertising budget of $800,000, faced enormous U.S. automotive marketing campaigns such as the one that spent $8 million in four months for a car called the Edsel. Even with all that "convincing advertising," the Edsel went nowhere. Buyers understood the differences and small thinking prevailed.

Perhaps success depends on how you think about your mobile business. Steve Case, founder of AOL, built the largest media company in the world on slow dial-up connections (less than 56 kbps), which 88 percent of people in the United States experience as their Internet connection in 2001. People find great value in basic technology. Journalists and analysts often chronicle the disadvantages of wireless technology. In their jolly jaded confidence, they assert that people need powerful devices and faster connections. But a shrunken graphical supercomputer with a powerful connection is not useful on the road; nor is it likely to solve mobile problems. It is likely to slow you down. Mobile users need on-the-spot personal information from handy wireless devices. They need less, not more. This is new territory and a development team that is knowledgeable and focused on wireless possibilities can help discover the best purpose of a mobile application.

Much of the world either has a PC or does not need one. Many parts of the world find success in the wireless Internet as smaller, more efficient wireless devices connect simply with institutions that supply content and services. The makers of the technology and the culture that uses it understand that small is beautiful. This direct thinking has led to generous adoption rates in Japan. Must we wait for the gas-guzzling, power-hungry vehicle when it is clear that a small Bug will do?

"Thinking small" is the basis for a new wireless culture that experiences good mobile applications. It finds that Web sites waste time; simple text makes the point; direct personal applications work best; information about location and personalized data is an advantage. Thinking small is about messages and notes, not

Table 24 Features of wireless devices compared to features of PCs

Wireless Small Talk	PC Big Talk
Battery life in days	Power supply in minutes for blackouts
Small and portable	Big and powerful
Weight in ounces	Weight in pounds
CPU functions in silicon	CPU functions in memory
Fits hand, fits pocket	Fits under a desk
Storage in megabytes	Storage in gigabytes
Memory in megabytes	Memory in hundreds of megabytes
Screen visibility, lines	Screen size 19" or bigger, millions of colors
Personal content	Enterprise application software
Touch screen	Three-button mouse with thumbwheel
Small keyboard	100 keys + 12 functions on 18"-wide keyboard
Network always on	Network always on
19.2 kbps or faster	128 kbps or faster
Coverage	
Voice clarity	

email and documents. The wireless features listed in Table 24 are for those who think small and talk small. A typical mobile traveler's feeling about mobile devices is, "The cool thing about my Web phone is that when I get off a plane, it sets the time automatically. It knows where it is. I do not have to boot it up. It does not crash. I can use it in more places than my PC. My important information is with me, as I need it. It is always on, I am always connected, and it fits in my pocket like car keys."

Miniaturization is certainly the key trend in digital device manufacturing. Some say that cellular phone screens are too small. While these developers wait for the screen to come, leading engineers see a minimalist opportunity to build mobile wireless applications that are small and powerful. It is a design opportunity that does not come often. In a small text space, one can communicate simply and to the point of necessity. The challenge for developers is to determine essentially what people want when they are on the go and arriving at specific places.

Experiencing Wireless Development

Wireless developers are quick – quicker than most Internet Web developers. When my business partner Roland Alden and I worked at Metatext Corporation, which made prepress software, we had a standing rule in software development: When you write software – even after an hour of work on something new – always leave your work ready to be shown. This is a great survival skill. Picture the outcome always; never leave a process undone. "Getting things going" is articulated

well in the book *Extreme Programming* (XP). XP deconstructs classic front-loaded programming software lifecycles. The goal is to make new technology products come to market quicker. XP authors Kent Beck and Erich Gamma explain the development process as a successive set of user stories immediately addressed with spikes. A spike, a rapidly developed software solution that technically solves at least one of the stories' key challenges, advances a project rapidly. This successive evolution of stories and spikes defines a product by first exploring what can be done, committing to what will be done, and then steering the body of work to a destination.[1]

Guiding the larger project requires great management of resources and timing. Initially the inventor's approach to see what can be made with available technology is to explore alternatives. To reach a large audience, however, many professionals are required to produce, distribute, and maintain a working product. The large-scale projects are similar to a film production model; a script and storyboard plan all the shots so that when the special gear and paid professionals show up they can rapidly produce the piece with little time wasted. Speaking of paid professionals, key wireless talent is required to produce the project and here is the cast of characters and their roles.

Wireless Core Team

Most of the work in wireless Web companies is done by professionals with wireless job titles and task responsibilities. One of the rarest talents that a wireless team manager can hire is the interaction designer (Figure 37).

Wireless team composition now includes a wireless architect, an interaction designer, a wireless client and server programmer, and a mobile content developer, in addition to people who perform other traditional software development roles. Because this industry is so content-centric, an editor and creative talent are also needed to keep people on the air.

A small team has the flexibility and speed to develop wireless projects. Certainly companies have complex wireless projects that require organizing an army of people to build and provision specific hardware, engineer special-purpose software, uniquely secure content, and provide special support and training. But the wireless market has no room for very complex systems that take a year to build. The technology changes are so dramatic that any long-term deployment is likely to mire an organization in an outmoded system. By and large, the wireless market is evolving to allow more of a mix and match of hardware makers, component software producers, and new content providers to work in shorter production cycles using tools for instantaneous distributed change.

Many companies that add a wireless channel to their Web sites get by with a small wireless team, although they can get help from the outside from those skilled in wireless ways. To do a first-class wireless job, your project will require

both new and traditional talents. The new wireless talents – the wireless architect, mobile interaction designer, wireless programmer, and mobile content developer – engage for brief periods during a project. The interaction designer should be consulted during the entire process.

The *wireless architect,* also called a wireless systems analyst, is able to assess an existing system and make recommendations about how to open wireless channels. Architects make high-level technology decisions largely based on current realities and pending developments. They keep an ear to the ground for the wireless industry changes coming in the near future. They are valuable because they track the wireless Internet industry and responsibly select a wireless architecture that permits anticipated growth.

The *interaction designer* is the most unique talent for a wireless project. These people are rare and are more often contracted from consulting or design firms. They understand both small screen design and mobile audience purpose. Understanding goal-oriented audience motivations, as well as processes for information design, persona, scenario, and storyboard development, is their specialty. Interaction designers own the "user experience" throughout the project. For wireless Internet projects, they observe and interview real mobile end users in the mobile environment. They are often active personalities, perhaps interested in drama and dance, and are physically available to work out in the field with mobile users. Successful projects allow these professionals to work with the whole team to keep them in touch with end users' needs during the development process. What makes interaction designers unique in mobile design is their understanding of action. They often go outside to work with the mobile audience using devices for their mobile information.

WANTED
Interaction Designer
for Wireless Applications

Looking for a creative interaction designer to work as usability specialist, committed to a career in wireless applications and new consumer technology interfaces. Opportunity to invent new applications involving location-aware services, personalization, time-sensitive and synchronized applications with emerging wireless devices. Has familiarity with Web applications and has designed interfaces for at least one client wireless device – WAP phone, handheld, pager technology in either industrial or consumer applications. Has designed mobile applications for at least a few of the following language platforms WML, cHTML, XML, HTML, PQA, Java. Can contribute to mobile information architecture and take full charge of illustrating storyboards and writing scenario-based narratives. Will construct mobile personas, and have mobile audience-profiling skills. Game design experience also worthwhile. Bonus if you have experience designing Bluetooth, GPS, SyncML, applications or have experience in messaging, microbrowsing, or interactive wireless applications for wireless portals. You will communicate as part of a team with both customers and engineers to explore new solutions.

Figure 37 A want ad for a designer with rare but essential talents

The skilled *wireless programmer* knows client and server development. Server-side work is often done in Java, although a Microsoft.NET language also works for Intel servers. Wireless device programming is often necessary and requires

familiarity with the software development kit and language for the device. This can be C++ or Java. Since wireless applications can run the gamut from messaging, to browsing, to interaction, these software engineers usually have a specialty. Messaging requires knowledge of paging, mail server protocols and push technology, and involves development on the server. If the messaging response is complex, then client software development is also required. To program for the wireless browsing model, the wireless programmer works entirely on the server where knowledge of the markup language and the dynamic binding to the database have to be developed. For an interactive solution, programming is generally synchronized between the client and the server. Specific wireless applications require skills such as game engineers possess.

The *mobile content developer* is aware that having your information on you is one of the key advantages of wireless. These developers participate in the information design along with the interaction designer. They can rapidly build prototype databases with realistic content to feed a prototype. Content developers realistically size the data and traffic requirements, generate database schema, use SQL databases, represent data as XML, can generate a DTD or XML schema, and advise how content can be presented dynamically through the server. They are concerned with data synchronization and representation on the wireless device and the server. Sometimes XML schema has to be created, but this requires research to work well with participating companies. Mobile content developers help partition data for partial mobile use. They often have to repurpose databases by adding location, time, personalization fields, or tables for mobile use. Security and privacy issues are also important, often involving collaboration with a wireless security engineer who works on the server. Content experts may also be consulted to make decisions on the sources and character of the data for mobile use.

Finally, for voice portal projects it is important to engage the services of *linguists*. They understand many of the finer issues of listening and speaking. Simple things like pauses, pacing, and diction help make a conversational application work well. If you are hiring, an ideal candidate is a linguist who has programming skills. A linguist with VoiceXML skills fills a similar role to a Web designer with HTML skills. The "out-of-the-box" voice portal uses a synthetic speaking engine. Although it works, it is not as warm and friendly as you might want. If you want to make your site sound professional, you need to hire vocal talent to replace and augment the computer-generated text to speech (TTS), with recorded speech and prompts. You will probably also need to hire a recording engineer and book time in a recording studio.

People with traditional software development talents also take on wireless responsibilities. The project manager understands and communicates the new wireless development method mentioned earlier. Managing the wireless process involves the internal development team and possibly external clients. The process sets the

schedule for dependencies, deliverables, and roles engaged in building the product. The manager coordinates deliverables from the development team. Day to day this involves the management of the progression of the mobile design, the implementation of client and server software, monitoring mobile content development, and championship of and sensitivity to the popular use of the new wireless channel. Certainly, traditional engineers are involved in server development and security issues. The quality assurance staff involved with an early view of the storyboard development can be very helpful. They will certainly give important feedback during the many tests and deployments of wireless devices. Developers can use third-party wireless testing organizations to offload the complexity of the testing process. Training is important because much of the world needs education on the limits and capabilities of the wireless Internet. Support is also useful to keep deployed systems operational.

The Process for a Universal Industry

Earlier we talked about getting things going. This is the right attitude of experienced professional developers: Let me at it. But to appreciate the universe, you have to step back a bit to look at how things come to be and how they go.

The fashioning of a wireless industry borrows from telecommunication, computing, and even publishing traditions where creative observations of cliché needs are manufactured into an archetype of utility. Telcos always talk about *plan, build,* and *operate*. Of course, the project has many steps. The "operate part" is new to many computer software engineers. The computer industry is used to construction cycles. Software development takes a project through *concept, prototype, planning, production, final testing,* and *maintenance.* Software engineers are often unaware of the people that actually use what they make. Yet an audience has ongoing technical needs that require continued engineering and development. Typically, when it comes time to build version 2 of software, an entirely new team is formed, cut off from the audience that is using the product.

The product itself has a lifecycle. I think the wireless Internet industry realizes that digital content benefits most from a strong participatory relationship between the technology providers and the mobile users. Telcos know that operation is the critical element of sustaining a business and healthy media publishers build circulations based on an audience that expects more.

Creative and Realistic Beginnings

A great idea or even a simple observation of needs can be the origin of a new wireless application. A playful SMS game, an interactive model of a city, or some utilitarian idea about the use of Bluetooth, voice, GPS, or other technology can spark a prototype that goes quickly to development and production.

Realistically, a project needs a technology and business concept to justify expenses before development can begin. Commercial wireless development often takes the concept or prototype and matches it with a wireless business model. The business-interested developer can study "The Site: Wireless Business Models" in chapter 16 to determine appropriate models. On the other hand, the wireless work may be handed to you via a request for proposal (RFP). This is a commercial statement that frames a concept to be developed, to which a development team responds. Whatever the cause, the project manager draws up a set of product requirements; however, this is often exploratory and requires some investigation of the scope of the project and the users.

During the investigation of project scope, some inspired ideas are generated; however, the purpose in the beginning is to understand mobile user tasks and gather customer requirements. Feasibility is quickly explored and hopefully useful discoveries are made.

A wireless architect makes early decisions including confirming the target audience and target devices and networks, and determining any third-party or internal tools that engineers need to build. They sometimes make recommendations about the process to ensure the ongoing life of the product through operation.

We are still forging a universal process for wireless development. Fortunately, the wireless development method is fairly solid. In the making of wireless projects, the wireless team moves through development phases in a special order.

The Wireless Development Method

The development method is special for wireless projects (see Figure 38) and resulted in very successful, quick production of wireless applications for Sony New Technology communicators and General Magic wireless vertical market applications throughout the 1990s. It involves the following key steps.

1. *Study the mobile user.* The interaction designer identifies and studies the audience. The mobile user drives the actual application with the interaction designer's creation of a persona, the primary typical user, and the scenarios in which the persona acts. A wireless architect then reviews the persona, the potential devices, networks, and messaging, browsing, interacting, or voice applications. The interaction designer determines suitability from a user's standpoint. The architect determines feasibility from a technical standpoint and produces a *wireless application plan* that serves the mobile user, while the interaction designer produces an information design for the content and storyboards that outline the flow of mobile activity.
2. *Create a mobile database with content.* The mobile content developer takes the persona study and application plan to develop an information design. XML

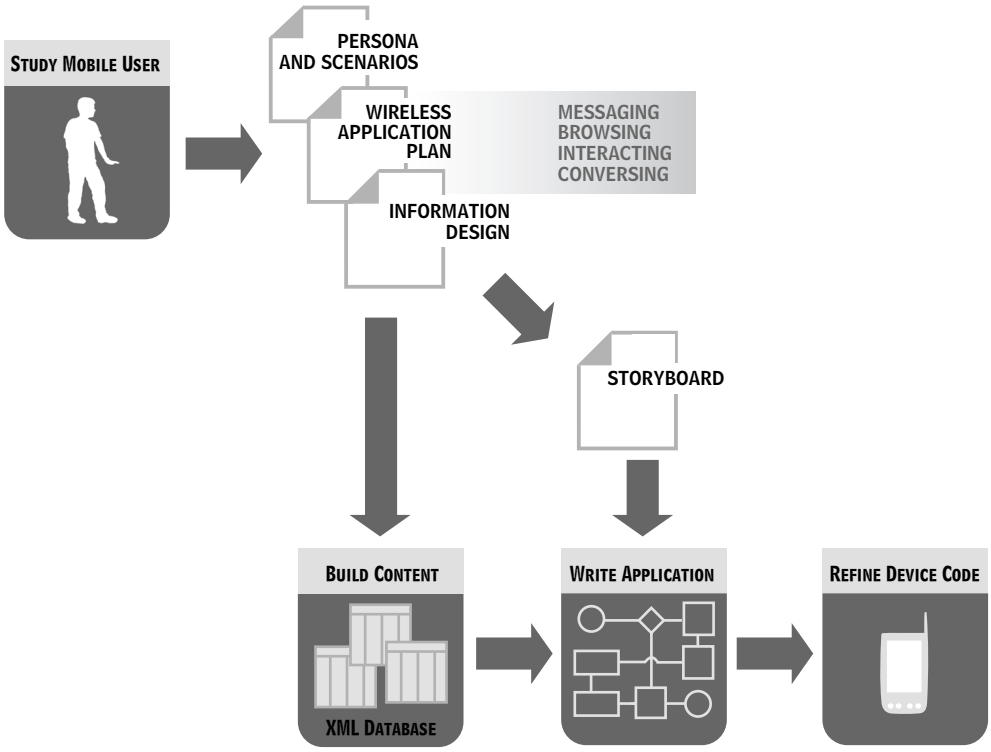


Figure 38 Wireless development method

suitability is examined for both the wireless application and server utilization. Content is loaded for use during early development.

3. *Design a logical application.* The wireless programmer quickly lays out a working model suggested by layouts drawn by the interaction designer. Universal to wireless is the next step. A developer fires up a simulator, builds wireless screens, and codes business logic while attached to real content.

   The major components in the application are:

   a. *Wireless field application.* The primary application for the mobile user on the go is the focus of a project. Many early wireless projects stop here and do not include the next steps, which are necessary for mature wireless applications.

   b. *Back office administrative application.* You may have built an incredible wireless application, but remember that someone must manage the mobile user content on the server. Some engineering thought must be given to building

useful tools with a good interface and to the necessity of administrative involvement to maintain a wireless system.

c. *Design a Web PC start page.* If it is appropriate to the persona and application, a personalized Web PC start page is added where the user sets up an account and personalizes the mobile application. This can let the mobile user administer the account and set up a mobile site to make it ready for mobile use.

4. *Use a device.* A consumer device using a real wireless network exercises the application. A good test uses the scenarios established when the persona was created. Ideally, the device is put into the hands of a real mobile user who matches the persona. Real content can then be generated.

The wireless project quickly gives you one mobile user, one working database, one application, and one wireless device to show off. But you take time continuing to enhance the content and the application. Unlike most software projects, wireless development often requires quality assurance testers up front in the design process. Experienced quality assurance professionals are aware of the potential mobile audience and help the interaction designer with insights into the persona profile.

You have noticed in the process that the emulators and devices appear later. Ideally, you have an open-ended wireless contract to select the correct network and device after you have studied the mobile user. Quite as often for commercial reasons, the wireless hardware and networks are fixed before the project begins. In either case, a wireless application plan is produced and signed off. You need to know what tools to get for your development platform. Tools are listed by device later. Ideally, each wireless device requires its own storyboard. The storyboard will show how content is displayed and how input and navigation work based on the characteristics of the device.

Wireless Demo

To show the world that small is beautiful, a demonstration or prototype is often required that is the basis for the final application. A prototype of the application, a demo, or a pilot test of a working sample is helpful to establish the working nature of the solution and show this example to investors or participants. This often takes the form of a demo that can inspire the beginning of project. A great wireless demo (Figure 39) largely involves stage production skills. Unlike PC demos where you can use a large projection monitor, wireless demos should move around with real devices and close-ups. It is important to avoid using a device emulator. It is more effective to show the application running with an occasional camera close-up of the real data appearing on the wireless device and changing on a PC "big

board." By big board, I mean a large control room–style information display that is being updated in real time as wireless devices make changes from the floor.

The complete business operational plan, completed designs, and a production schedule are required. This happens some time before making a prototype and definitely before production. Just as in making movies, wireless development requires a well thought-out storyboard and a detailed set of working plans. A solid requirements plan and architectural description can save a fortune.

In production, professional application and content teams write large amounts of code and collect or repurpose content. With a plan, teams go about their work on devices and servers, and content is often generated during production. Without a plan, production can come to a standstill as people go in different directions. The software cycle approaches completion with rounds of testing that include ever-larger audiences. Finally, the product can be planned for distribution where any steps for provisioning are made.

- Easy-to-explain overview
- Real devices, not emulators
- Real wireless network
- Active use cases
- Live stage presence
- Big board view of changing wireless transactions
- Serious content
- New datatypes
- Multiple wireless users
- PC Web interface tool
- Elmo or wireless camera for close-ups.
- 5 minutes or less

Figure 39 Recipe for a great wireless demo

Some large organizations actually provision the devices and the wireless service for their customers. *Provisioning* makes available to a customer a tested package including the right wireless device, compatible peripherals such as memory cards and earplugs, the right version of the software, and any account setup. When the customer receives the shipment, it works right out of the box. Wireless provisioning is a large task that can include hundreds or thousands of users each month.

Wireless Developer Best Practices

There is a saying that best practices are used because no one paid an engineer to figure out how to do it right. *Best practices* are professional guides that help professionals avoid doing things that waste time and money and help them produce high-quality products. Recall the wall of wireless complaints in chapter 2 (Figure 10); you want to resourcefully overcome objections of early wireless technology. To avoid those traps during the course of development, experienced developers keep in mind a number of general lessons to improve the quality of their wireless applications.

- Design mobile database and load sample content early.
- Use interaction design to minimize navigation and eliminate unnecessary tasks.
- Use information design to simplify content.

- Use big fonts as often as possible.
- Simplify the content with unformatted text in short sentences.
- Design black-and-white composition first. Gray, not color, screens are the first market.
- Explore mobility. Play with portable information. Forget the PC Web model.
- Identify core business utility and key transactions.
- Characterize only the mobile audience and get early participation.
- Convert all input text fields to choices and buttons, if possible.
- Save the user's time.
- Review for consistent navigation and layout.
- Do not mix interfaces (i.e., voice, gesture, keying text); stay in one medium.
- Allow for misspelled input.
- Make error messages clear, helpful, and actionable.

If you resourcefully steer around limitations and provide a meaningful implementation, then users say good things about the wireless Internet.

Defining Your Mobile Audience

It is important to clearly identify a primary user and the primary device. As we work in product teams, and concentrate on deadlines and flowcharts and build wireless applications with new technology, it is easy to lose sight of the end customer. Although many wireless engineering teams forget for whom they are making products, they have the ability to liberate the user. A usability specialist or an interaction designer is usually in charge of developing a profile of the end user to help a wireless team recognize the mobile user and individual needs.

It is important to feel that you are building for someone real. Andy Hertzfeld, co-inventor of the original Macintosh Computer, suggests that you build something new for yourself first. Care about your invention and think about improving its purpose to give your life advantage. If you take your work to heart, you might actually start dreaming about it, rather than have a nightmare about avoiding it. Pride comes with making something useful. But you really have to take this observation further; otherwise, you will build something that only an engineer will love. When you move to the goal of building for other people, creating something for a business person or an average citizen, it is important to realize that you are not one of them. You need to set someone else up as the model to use your invention in order to make sure you are not just kidding yourself. This model goes by the name of persona. Let your persona dictate the necessities that will be the mother of invention.

The technology and the mobile audience for wireless applications are very new. They are significantly different from what has come before – for you and everyone on the team. A successful practice that informs developers and qualifies the audience is the identification of the mobile persona. There are three basic steps to studying the mobile user.

1. Develop a persona.
2. Describe scenarios.
3. Create storyboards.

From this work, engineers can build products. Jeff Raskin expresses a working sense for mobile usability when he says, "An interface is humane if it is responsive to human needs and considerate of human frailties."[2] Wireless interfaces need to be both useful and able to recover quickly from human errors. It takes a special sense to know how to design so that the right wording and the right functions can be used, and so that recovery from their misuse can be graceful.

Personas

As we have said, the mobile user needs to be defined. There is some confusion about the identity of the mobile end user. One of the causes for bewilderment is that the kinds of users differ with respect to the maturity of the technology. The point of the technology adoption curve is that you begin defining your audience generally as hobbyist, business, or consumer by knowing the point in time of your technology. You apply that technology to the appropriate audience. One surefire technique that clarifies the audience and gives you important development tools is the building of a persona.

A *persona* is a single, concrete characterization of someone who uses an application. There can be three or four of them, but a project always needs at least one primary persona. The creation of persona involves the identification of the typical personality, the writing of a background, a description of characteristics and actions that define the "presence" of the personality, a photograph or illustration of the person, and a "day in the life" of the persona that shows typical activities. This is not a real person, and the picture should be anonymous. It is important that this persona be introduced without explicit reference to your proposed technology. Later in scenario writing, we envision the persona using the wireless technology.

The invention of a "wireless mythology" of a mobile user performing actions is the art of building personas and scenarios. The persona's spirit lives throughout the project. Once a persona is defined, traditional development questions are often resolved by simple reflection and discussion, and answers to the question, "Would Maria (the persona) like this feature?"

An interaction designer is usually in charge of constructing personas. A brief persona description and sample scenarios together guide the wireless project during design and development and are often delivered to a client along with the design deliverables.

A persona is not a real person but a specific characterization of a class of people and their usage patterns for a particular application. A persona must be embodied in a realistic depiction, however. For your wireless application, personas become invocations of the use. They can inspire team members who are often energized by the mascot mythical figure while the project is being developed. This hypothetical mobile user is the result of a merged set of demographic identities for which you build the wireless application. Once constructed by the interaction designer, the persona provides a realistic quality for the wireless team and the client. A persona, although fictional, has a realistic personality. At a minimum, the profile for the persona should have:

Figure 40 "Maria King" persona. (Photograph by Mark Beaulieu.)

- *Name:* Maria King
- *Title:* Senior Account Executive, Navco Manufacturing
- *Picture:* See attached (Figure 40).
- *Biography:* 32 years old, MBA, mother of 2. See attached.
- *Personality:* Likes profession. Reads *WSJ*. See attached.
- *Goals:* Gain corporate visibility. Improve abilities. Show bottom-line value of all technology.

A short one- or two-paragraph biography for the persona is important. The first paragraph of a persona introduces the person and his or her life, without reference to your technology. It is fair to say that in the early phase of the wireless industry, no one really knows the fundamental qualities of the brand-new technology or what the mobile user wants.

Goals and Tasks

Goals are key elements of personas. Goals spring from the persona's identity. Goals are personal and often embrace professional objectives. Once goals are stated, a design team can help the end user achieve them. It is part of the fun of the mobile design challenge. One look at military wireless mobile goals can be an eye-opener.

Soldiers who use mobile wireless technology have very definite goals. Recently, the U.S. Defense Advanced Research Projects Agency (DARPA) sponsored the development of the Situational Awareness and Information Management System (SAIM). The system feeds wireless battlefield data to soldiers. The

program is intended to create an "information bubble" to serve soldiers in action. "On the battlefield, you don't have time to browse," stated Prasanna Mulgaonkar, SRI International lab center director on the project. He stated there are four fundamental issues that a soldier wants to know at all times:

1. Where are my buddies?
2. Where am I?
3. What's going on?
4. What do I do next?

The litany of mobile battlefield goals of a soldier transcribes to civilian activity. The coordinated needs of a mobile field force often have the same goals. Everyone needs location and direction.

The needs of a person on the go are certainly different from the person sitting at a desk. The early success of the Palm handhelds largely comes from the company's respect for the fundamentally different needs of the mobile user. Therefore, when developing wireless projects, think of the persona as the spirit that builds your application. You write the code, modify content, and deliver the service to this persona. Working for the mobile persona and being in touch with his or her content is how great applications are written.

It is likely that each of your projects will be for a different kind of end user. The personalities change from the experimental technical crowd, to the ordinary business professional, to the liberal early consumer, to the easy-going mass population, to the late-adopting conservative where nothing can go wrong. Development teams that go to the trouble of defining mobile personas often find projects exceed the project's stated goals. The design firm Cooper Interaction Design makes a very strong practice of goal-oriented design. Its site is <http://www.cooper.com>.

Studying Persona

Look forward to starting new wireless projects with one mission. Find the elusive mobile personality in the act of achieving goals. Like a detective, you observe and write down notes about real end users as they complete central tasks. You want to distill the successful qualities of this persona. Tasks can be derived from goals, and are prioritized by them as well. Mobile users are the true clients. A good way to begin is to become familiar with customer demographics. Read users' letters and understand surveys to get an idea of the baseline of the audience. Notice their jargon, their interests, their short-cut notations, their language, their imagery, their concerns, their intent, and their goals. Pay attention to their words and the order in which they do things. Observe their locations, time-sensitive needs, and personal activities. You will see a real difference between building for an installed PC base and building for a new pool of mobile users.

The investigation of the persona often requires interviews. Here are some key questions that are answered in this book: What content interests your audience? How dynamic is this content? Does it require heavy user interaction? You also want to find out when and where they will use the data. Are they at home, at work, traveling for business, or on vacation? You also need to determine the networks that are available geographically. Will the audience remain in a specific area or travel among regions? If they are traveling, what networks are available at home, at the destination, and en route? All these questions were answered for the Nokia mobile user study in chapter 1 in the section "European Wireless Banking."

You want to get a broad feel of the user base to characterize audience by their personas. Understanding identities and personas is key to giving your product personality and purpose. If you do so, your product may not work for everyone, but it certainly will work for these people. You should collect the few key personas you need, but you and your team then need to reduce the collection to one primary and no more than two, possibly three secondary personas. Over time, the persona study will become your jargon, your notation, your language, your imagery, your concerns, and your goals.

Your job is to create a purposeful network application for personas. Once you have identified them, you put into place the simplest mechanisms for the persona to achieve his or her goals. Beyond the generic persona, you can add customization to help end users designate their identities. People signify their personalities with the details of content and you can often take advantage of local time and physical location.

Although it may be fine to think of your end user as a consumer, that is not enough definition to guide your application. You need to come out of the fog of marketing demographics and into the light of a crisp characterization. You may need a skilled writer to help you write a simple narrative. A persona needs enough detail so everyone has a feeling for this person. Turn the average mobile corporate user into Maria King, Field Account Executive for Navco Manufacturing. Her goal is to be visible in her large company and her tasks are to keep its customer banking networks operational and give the best service she can. She covers the San Francisco Bay Area field accounts for the ATM line of her banking products and services. She spends most of her time upgrading network products. She is also mother of two and has a Web phone to keep in touch with the family as well as get updates from the field as schedules change. She cares that her phone is dependable and keeps her in touch reliably. She turns off her phone so as not to be disturbed. Between meetings she checks voice mail, completes decisions with the company messaging system, and sets up her meetings for later in the week.

For vertical market handhelds, your average field professional becomes Bob Hurley, Field Service Technician for Aimless Networks. His goal is to be a productive installer and not work overtime so he can enjoy his family life. He drives a

company Ford van six hours a day managing work, taking a regular lunch, and getting back to the family for dinner. He installs six network converters a day and uses his phone sometimes to borrow inventory from nearby drivers. He uses a truck-mounted detachable handheld that has voice. He checks off his installations and when he docks his device, it automatically recharges and updates the dispatch list after every call, alerting him only if a change to schedule takes place.

You will make discoveries as your personas teach you what works and what does not. Users think in terms of the problem that they are trying to solve, and not in terms of how to use the tool to solve the problem. The best tools are transparent to the users. Whenever users have to stop to figure out how to use the tool, they get sidetracked and frustrated, forgetting the problem they are trying to solve. Having an eye out for the customers as they go wireless with your application, you can often observe common patterns of use. With willing users, you can explore with them how best to save time. As much as possible, reduce instrumentality, and cut out what you can. Observe user repetition, what they consider as favorites, and come up with ways to "remember repeated patterns." There are obvious points in software where users should not have to type in information that is well known. Make capsulated "one click" buttons of operations. Streamline steps. If choices get too complex, consider laying them out on a Web site for customer preselection before going mobile. This is a successful strategy of the mobile portals from Oracle or Yahoo!

The Finer Points of Persona Development

Focus groups that look at a project after release often reveal that technology was not being sensitive to real users in the first place. Sometimes focus groups are needed, but try to avoid them. Instead, do your real user focus work up front. Again, before any development, be sure to model users as personas. Two major qualities come to mind in defining your audience – strength of nerve and strength of imagination. It is easy to be overwhelmed when you observe so many mobile users, work with differing client personalities, and are busy developing your team's various technical and creative talents. It takes a lot of energy and inspiration to conduct the development process. Even with great ideas, you have to make hard decisions to be productive. Be bold; have no fear. Conventional wisdom often fails to exploit new technology. It is important to either inculcate creativity yourself or have trusted people on your staff who are tuned into the shape of things to come. Insensitivity to the imagination of your audience is a cause of many lost opportunities.

You should get to know your audience. You may know that you will support active interfaces and construct direct designs. But personal detail is helpful. Begin by reading client demographics, users' letters, and surveys. Understand identities and personas so you can get a feel for the breadth of your solution. Read their jargon, write their notation, use their imagery, and live in their world enough to

sense their needs. Meet real users and profile their goals to help characterize the audience as personas. You want to have a real "Maria King" tell you how doing some wireless task truly saves her time. The motivation is not so much to help you figure out an engineering solution, but to put you in a position of continually feeling good about helping people live better lives. In any case, field testing an application and noting actual behavior are great tools for identifying possible usability improvements and alternatives that improve an application. For more information on this subject, see the section "Demographics, Profiles, Personas, Identities" in chapter 21.

Voice Portal Personas When it comes to developing voice applications, the persona study does not change. However, it is worth noting the points in the persona's life where he or she has time for conversation. Voice portal applications can start as dialogs outlining subject and flow. Later, refinements for natural speech and alternative forms of dialog can be developed. As mentioned, linguists are very useful in the refinement of the project.

Using Professionals as Personas and in Your Project The ability to rapidly construct meaningful personas and useful scenarios is as important a resource to a wireless project as a good engineer. If you can hire an interaction designer or a design firm, they will save you a fortune in downstream project expenses. Although most of their work is done up front, having an interaction designer and project manager advocate the personas and their needs during the course of development can short-circuit many "misinformed" engineering decisions. Beyond personas is reality. If you do not have an interaction designer for persona creation, because having a full-time person may not be cost effective, it may be useful to contract for this service. Useful examples of persona development appear in chapter 9 of Alan Cooper's book *The Inmates Are Running the Asylum*.[3] To help you learn from actual users, quality assurance experts provide excellent end user feedback. Employing seasoned quality assurance professionals early in new technology projects can also be very helpful. Usability experts sometimes apply sophisticated techniques to verify their interfaces, as we discuss in the section "Measuring Interfaces" in chapter 21.

Relevant Content Personas may want personalization, locality, and time relevance, but they always need a navigation model that makes sense to their world. A multimedia interface designer once designed a kiosk interface that would present a library of fish to fishermen. She came across an idea for a conceptual model by observing her audience. Every fisher knows that fish are drawn to favorite stream locations – fishing holes. She used the fishing hole as a navigational and retrieval model. The kiosk was very popular. Unfortunately, thoughtless models are easier

for careless engineers. I am thinking of a wireless traffic-reporting system. A group was given the goal of providing mobile traffic reports to wireless WAP commuters. The engineers simply took the traffic streams and presented the entire list of bulletins for the audience to browse. The engineers did their job and in record time. But this delivered only raw content from a traffic-monitoring agency. The content did not match the world of the driver. Without thought given to organization and relevance, the content will not find its target. Drivers had to repeatedly find what was in their area by accessing bulletins to see if the areas they were traveling through were mentioned. This wireless service was too much work for a driver and had no mobile following. The solution needed a more personal model based on the driver's world. If content is to be valued for being location-based, time-sensitive, and personalized, then how better to find out what this means than by spending time with the people who use it and care about it?

Creating Scenarios

A *scenario* is a concise description of a persona using technology components to achieve a goal. The scenario considers how the mobile user handles the hardware, how the application is operated, and how the content is used. There are at least two primary types of scenarios to consider: the daily use case and the necessary use case.

Daily Use Case Scenarios

It is important for a team to list all the possible requirements for a product, but the chief focus should always be on the central actions the person is doing with this product most of the time. Functions that are used every day should be handy. Products often make the mistake of providing all the possible functions up front. If many daily operations are used, then design a progressive disclosure of functions. Hide away the sometimes used functions, but put the most used information and functions in a space that is easy to get to. Signify location, time-relevant, and personal needs. Define where and when the application is used. Observe people acting out their data. Europeans have done a stellar job of this. By doing more with less you can see successful WAP and i-mode applications where the user moves from an airport to a taxi to a hotel – checking in, planning an evening, and so on. This solution can involve related wireless companies, for example, those that complete wireless transactions.

What a person does most often in a wireless application should be presented foremost in the interface. Making phone calls or reading email is a common wireless activity. Supporting the most needed functions is a primary design task in building the wireless application. Reading a daily use case scenario, one should easily see the most used functions actually solving the persona's goals.

Necessary Use Case Scenarios

As mobile users need less-common features, you provide them in a place that is not on the main path of choices in the interface. Complexity is best progressively disclosed to the user. For example, reading email is on the main path because it is used every day. Replying to email is on the main path. Composing email is not on the main path. This capability should be a short step away from the main path so it can be found and used when needed. The users should be able to figure out how to do it without external help from other users or manuals.

When composing a scenario, meet real users and profile their goals. Use their language. Learning from this approach means moving around while using actual wireless phones, not simulators.

Creating Storyboards

With the scenarios in hand, and sometimes with a conceptual sketch from the wireless application plan, and an information design outline, an interaction designer can draw a storyboard. A storyboard diagrams the entire story of use, one screen at a time, and shows the display, navigation, and interaction for the wireless device. There are three things to keep in mind: the mobile user conceptual model (process), the display the user sees (output), and the command system through which the user operates the system (input). The screens are laid out in order, with navigational options indicated.

Getting to know your eventual user is the essential beginning of storyboards. It is also worth consulting a wireless architect who can illustrate the general sequence of use for a wireless device and network model that will fit your case. Complex wireless projects have multiple storyboards, but it is best to focus on the storyboard for one primary wireless device. When you get more experience, you can consider the major and minor devices – the primary work going into the PC that has a high screen density and a pointing device and the mobile screens that have low screen density and some standard command buttons. As you will find out, presentation, input, and navigation vary greatly among mobile devices.

Figure 41 shows the storyboard for WAP email application login, including the key navigational Web phone screens. At Lutris Technologies, where this application was made, the storyboard was taken from a high-level site map that outlined the pages that would appear within the WAP email system. Note how attention in this storyboard is spent not on graphics and colors, but on fields and links.

Tools for creating this storyboard could be PowerPoint™ or Visio™, which let you rapidly lay out the text and graphics of the application. PowerPoint is useful because you can embed fields and links to simulate the application.

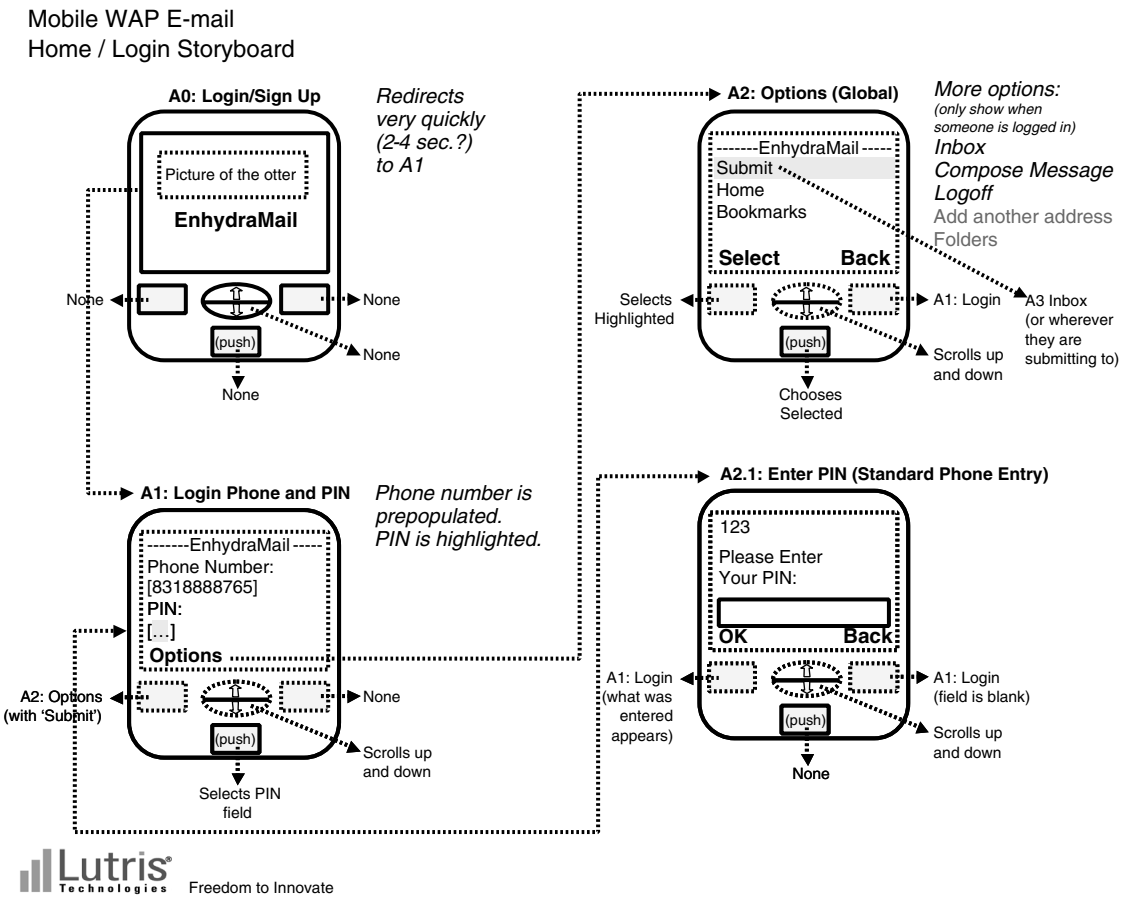


Figure 41 Storyboard for WAP E-mail application login (*Source:* Lutris Technologies, Inc. Reprinted with permission.)

Activity Diagrams with Storyboards

A PC-style storyboard approach is often a problem. Mobile users do not like to navigate hierarchies. They are used to a flow of control. To help advance storyboard design, developers often provide activity diagrams for the mobile user application, as shown in chapter 21, Figure 109 "Activity diagram for a wireless location application."

A wireless system can require two storyboards – one for the mobile field persona and one for a business console persona. The console provides means to administer the account, set up personalization, or prepare content for mobile use. The administrator usually needs only an HTML PC screen. The mobile user

clearly needs a storyboard for mobile use. He or she may also need an HTML PC screen to set up mobile preferences.

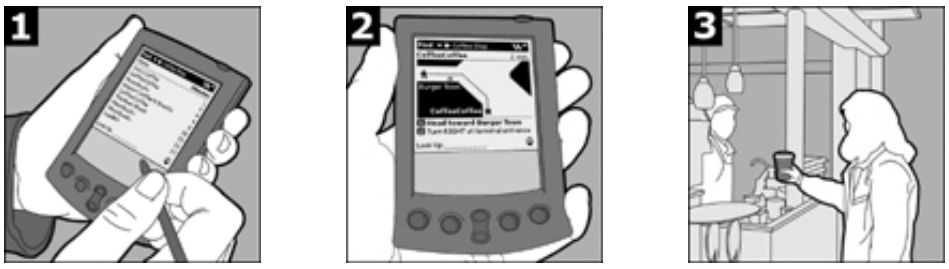
Of course, there are four wireless mobile families to design for: Web phones, handhelds, pagers, and voice portals. Each needs its own storyboard. Different types of devices sometimes require different types of storyboards to accomodate different features such as screen size, functions, and interfaces. For large projects, each family has variants; for example, a Web phone design is necessary for both WML and i-mode. But the best strategy is to focus on one device and make it work well first. The general navigation can be transposed to other mobile platforms, although there are many fine-grained decisions to make along the way. You must show the primary distinctions of the wireless device on its storyboard.

Web phones can do HTML and handhelds by Palm can do WML, so decisions need to be made up front about appropriate markups and approaches. Ultimately, it is less about the language you use for your device and more about the presentation and input. The plan and the storyboards must be definitive. To reach a truly global audience, you will have to design custom presentations for the families of devices. For an American audience, a PQA is often the best answer, especially if the application is complex and requires a lot of user interaction. For the Europeans, the solutions are more often SMS and WAP.

A handy usability statistic is, the maximum tolerable wait time for a mobile device is one-third of what people will tolerate for a PC. When the user is in motion, responses must come quicker. And a skilled wireless designer will make some decisions to decrease the number of steps. Of course, more steps will be eliminated in testing. Storyboards can go through a round of "paper testing," a practice of quality assurance personnel who take each scene and run it before potential users to evaluate flow and consistency. Storyboards finally go to the engineering team to implement.

Location-Based Scenario

The Wayfinder Scenario (Figure 42) is another example of persona development. Passing through airports, people often have difficulty finding the services they need. The availability of these services, as well as how to get to them, is not obvious. Enter our new persona, Angela, a 31-year-old public relations consultant who is based in Los Angeles, but who has to travel during the week to visit customers. Her *goals* are to be on time always for client meetings, to travel without hassle, and to not feel stupid. Her *scenario* is: Angela is on her way to Seattle and has a 30-minute layover in an unfamiliar airport. She wants to grab a cup of coffee before she heads to her connecting flight. The storyboard shows that on her handheld, Angela uses the list screen to look up the service she wants (Figure 42, board 1). After choosing a service, Angela sees the map screen (Figure 42, board 2), and



After Angela arrives, the airport map and service details are downloaded to her PDA via a wireless local area network. Angela quickly finds her favorite coffee shop in the list.

The Wayfinder shows Angela the nearby coffee shop and exactly how to find it with handy landmarks indicated on her map.

Angela follows the directions the Wayfinder gives her, and successfully finds the coffee shop. Soon she enjoys a double-tall, fat-free mocha latte grande.

Figure 42 The Wayfinder Scenario (*Source:* Courtesy of Cooper Interaction Design, Inc.)

then a close-up of the map (Figure 43). It shows her position, her destination, and major landmarks on the way. She can navigate by looking at the map or by following the simple directions to her destination (Figure 42, board 3).

The Wayfinder's design can be validated against Angela's goals. To meet Angela's goal of getting to her meetings on time, we made the Wayfinder interface simple and quick to use, with only two screens. To help her travel with as little hassle as possible, the Wayfinder includes a complete service database of the airport she is currently visiting, with reliable directions and information. Note the critical role of content to this application. To ensure that Angela never feels lost or confused, the idioms of the Wayfinder interface are familiar, consistent, and easy to use, even in an unfamiliar airport. The Web site for the continuation of this scenario is available at <http://www.cooper.com/concept_projects.htm>.

Figure 43 Wayfinder close-up of Figure 42, board 2 (*Source:* Courtesy of Cooper Interaction Design, Inc.)

Using the Wireless Application Plan

If your team is new to wireless development, then a short conceptual sketch showing all the wireless components can help everyone out. This *wireless feasibility study* is usually made by a wireless architect or an experienced engineer. They know the correct notation and familiar text. The conceptual diagram illustrates the mobile user, wireless devices, wireless networks, a server, and content sources. It shows everyone a first idea of overall use – the flow of the application and the content movement, as well as a sense for the number of devices to acquire for

development. General drawings of both the end field use and the administrative operations should be made. The conceptual diagram is brief. The conceptual diagram is not the "day-in-the-life" scenario, but a quick technological explanation of the system. It illustrates general operations and serves as an overview within the wireless architect's *wireless application plan*.

The wireless architect formalizes all parts of the feasibility study. While the storyboard writer will refer to the study, the wireless architect refers to this diagram and adds the correct layers such as the details of security or synchronization. The architect further illustrates how actions originate and responses are generated. As an alternative to the narrative text in the feasibility study, a timing diagram can show how data moves through the system over time.

When you are starting a wireless project, it is important to understand what you will use. A wireless application plan includes the special qualities of the primary device hardware, the presentation and interactive parts of the software, the specific wireless network, the nature of the data, a solution for how the content is to be used and managed, and the software tools you will use.

The network for the wireless application and its mobile database require construction notes because they must work together. As we will see in the coming chapters, the approach you need to take when developing for a WAN network is considerably different from the approach you take when developing for a LAN network. The wireless network protocols require different efforts to handle data and ensure security. The Web phone handset requires constant connections and has a much more compact interface, providing for greater mobility and lighter content. It deserves a direct and simple interface access. LAN devices are much more like computers, have greater capacity, large screens, and high data exchange rates, and can work offline.

Sometimes with your wireless application plan you can invent wireless applications, at other times you are responding to the market. Regardless, the application must match the maturity of your audience. Recall the technology adoption curve in chapter 2 (Figure 9). Applications historically mature with the declining technical sophistication and increasing service expectation of the audience. An early market application such as a youth portal makes sense. Early vertical market businesses match individuals with directed purposes. The quality of a wireless travel guide suitable for the early-adopting business person is different from that of a consumer variety.

In planning for specialty functions of wireless applications, you must anticipate further tasks. Proximity, location-based service, time-based tracking, notification, and alerting are valuable in a mobile on-your-person device. To get these functions, you may have to produce or buy wireless technology tools. You need to consider not only the wireless application, but also the day-to-day operation for most wireless systems that involve content. Other tools and components required

for wireless development that support text messaging, personalization, location, calendars, and other content include servers, gateways, and special engines. They are covered in chapters 18, 19, and 20.

Feedback: The Beginning and the End of Testing

Wireless connections may operate in one city and fail in another. It is best to test in a variety of physical locations. Conditions vary widely in wireless. For instance, you may be able to simulate the loss of a connection between modem and basestation by removing the modem's antenna while it is communicating. Or you could move your application out of your coverage area as you use it.

If you are using middleware that employs a server at a central site, test it under the load of multiple mobile stations as they simultaneously access it. Developers find differences in the ways operators process the signal.

Final validation of a wireless solution means that you must test the solution in the primary use locations. Wherever you expect your primary population to use the application, testing and refinement for deployment are essential. For example, if your wireless application is expected to be used along interstate roads, then make sure the towers are there to support it with the devices you are recommending. Another example is a congested airspace like in New York City or Los Angeles. You may find when you are testing the air that TDMA or CDPD circuits are in fact saturated during the expected peak hours of use. You may need to recommend switching to another data network and set of wireless devices to satisfy local customers. The actual traffic of devices affects the way you optimize your server.

The best Internet developers have made the medium work for them by involving the audience in the solution. Often real users make observations and recommendations about the service. Many changes can be made in short order without costly redistribution. The speed by which a site generating content can reach users who can then respond to it creates a feedback loop, which is unique to the medium of the Internet. This is not merely an editorial loop; it is a service loop, and that can lead to an evolved system. A resourceful development team can quickly post a wireless Internet system and, with skilled design, make changes to improve it while the system is on the air. This means that many a wireless system can be brought up in short order, thereby delivering value to the audience without having to wait the traditional cycle for a full battery of tests to be signed off, delaying market availability and early to market advantages.

As the wireless service becomes maintainable, it is important to provide a feedback mechanism where customers make useful suggestions for the general benefit of the network. You have effectively created a wireless network channel that connects mobile users in such a short feedback loop that response and change are immediate.

Exploiting Mobile Operation

It is very important to have someone assigned to observe the mobile customer periodically during the actual use of the wireless service. This person observes usage, detects unexpected values, and voices evolving goals of the mobile end user. The focus is no longer on planning and building, but on operation. Looking at how editorial, administrative, customer service, or customer support units operate in an organization provides some examples of mechanisms that are necessary for any business to survive and cary on a relationship with its "audience."

This follow-up not only shows you care about the customer, but that your organization is ready to provide industry leadership in what may be a new and legitimate business opportunity. Internet Web sites are able to do this all the time because their business is electronic and easy to change. As you seek to qualify your wireless network, keeping a flexible resource in reserve is key, thus permitting the wireless architecture to grow. The total value cannot always be foreseen.

Although Internet Web sites are able to change in short order, changing navigation and placement on a small wireless device can be very disorienting. This is because mobile applications use a command system that is learned by the mobile user. It is recommended that a pilot audience test system changes before issuing releases.

Innovative companies hold in reserve the resources to grow after deployment. It takes observation and intelligence to recognize how your applications are actually being used. More than once, consumers have been attracted to qualities of an application that had nothing to do with the original engineering plan. The trick is to care about the life of the product and exploit the serviceable value. Your company may let you observe and make changes beyond the initial release. Staying in the loop with the mobile user is the best way.

Wireless Development Tools

You can lay out and test an interface design in the presentation mode of PowerPoint, and on Apple computers with HyperCard and SuperCard. However, there are special third-party wireless development and design tools available and coming to market. Wireless handheld and handset companies like computer companies offer an SDK. They sometimes include it in an integrated development environment (IDE). These tools let you write applications, call operating system functions, and round out the software development production process. Every successful handset and handheld maker offers an SDK and support program, usually with the cooperation of a software tools company like Microsoft or Metroworks. Many devices, like those of the Symbian family, can be programmed in multiple programming languages. But not all runtime programs you make will work within the popular operating system. For example, although you can program a Linux application for a popular handheld, your end user may not want to

install Linux because it takes some programming skill. It will also kill any of the other PocketPC applications.

Today's popular wireless programming languages are C, C++, and Java. Writing code on an SDK must be tested on the target device. To speed this up, the SDK provides emulators that simulate the device on a PC. Many emulators let you debug the device memory model from the PC as well. Within the operating system of the wireless device are libraries of code with special application programming interfaces (APIs), the most obviously unique being the phone call interfaces for the telephone. Well thought-out handheld devices offer libraries that include communications, data handling, and synchronization. They provide a mobile interface and mobile components, and some define a high-level application control. The most advanced tools are Rapid Application Development (RAD) tools, which can save weeks of time. If the manufacturer does not have a RAD tool, check the third-party tools. We will look at many of the best tools in the next chapters.

Foundation tools are sometimes required. And you need good engineers to build them. Foundation tools are core pieces of software that let you produce other parts of the system and end up helping to make the entire wireless system work. This may mean having to write modules of code for your device, which could include memory managers, XML parsers, display components, Bluetooth drivers, mapping modelers, and record synchronization subsystems. It could also mean having to build special system tools. For example, for your application to deliver wireless content, you may need tools for the content experts. These back-office tools help collect source information, "clean up" odd pieces of data, and maintain the integrity of the content. It is worth pausing to consider if they are necessary. They do take time to build, however, they often improve the quality of content or save time when the system is deployed. Of course, you can spare engineers the task of inventing tools by checking on commercial as well as Open Source suppliers. You can often download trial versions of tools. Then you can determine if they are good enough or if engineering needs to go the extra mile to create special quality tools.