Bringing the Wireless Internet to Mobile Devices

Transcoding and Relational Markup Language are promising middleware solutions to the problem of bringing Internet content to the extremely diverse and dynamic mobile wireless devices universe.

Effectively mapping Internet content to mobile wireless devices requires not only new technologies and standards, but also innovative solutions that minimize cost and maximize efficiency to the benefit of both content providers and consumers. The wireless Internet must deliver information to handheld device users regardless of where they are and how they are connected, and in a suitable format—a challenge complicated by the dizzying array of devices, wireless standards, and applications.

SUBHISI Saha
Mark Jamtgaard
2Roam Inc.

John Villasenor
University of California, Los Angeles

UNDERLYING TECHNOLOGIES
A complex, interlocking set of technologies underlie wireless Internet services and devices. Equipment providers such as Motorola, Nokia, and Siemens produce the devices and the infrastructure to support wireless data networking. Microsoft, Palm, Symbian, and other companies provide operating systems and microbrowsers for handheld devices. Application platform solutions from vendors including Openwave, Nokia, and Ericsson contribute the middleware infrastructure such as WAP (wireless application protocol) gateways. Middleware also includes a new generation of wireless application platform infrastructure software to provide wireless applications and device independence for the increasing variety of handheld devices.

Network technologies
Most digital cellular networks are second-generation (2G) networks. A complicated set of overlapping, mutually incompatible 2G and 1G (analog) standards exist in the United States, while the Global System for Mobile communications (GSM) technology is the most prevalent standard in Europe. The maximum data rate in most 2G networks is 14.4 Kbps or lower.

The next several years will see the rollout of 3G systems in the United States, Europe, and Asia. 3G will initially support data rates in the tens to hundreds of Kbps range, with possible future support for data rates as high as 2 Mbps—most likely for low-velocity motion and short mobile-to-base transmission distances. Intermediate 2.5G solutions will leverage much of the existing network infrastructure and offer data capabilities in excess of what is available in 2G but short of the eventual 3G speeds.

Although the exact form in which these various high-speed wireless data services will develop is uncertain, we will clearly see substantial improvement over today’s data rates in the very near future, thereby removing one of the wireless Internet’s most significant hurdles.

Service technologies
Another obstacle lies in the mobile devices themselves, which typically suffer from small displays, limited memory, limited processing power, low battery power, and greater vulnerability to inherent wireless network transmission problems. These usability challenges make supporting common Internet standards such as HTML, HTTP, and TCP/IP difficult because they are inefficient over mobile networks.

WAP. To address these issues, a group of leading wireless and mobile communications companies developed the wireless application protocol for transmitting and presenting wireless information and telephony services on mobile handheld devices. Whereas HTTP sends its payload in a text format, WAP uses a compressed binary format for greater efficiency. It offers a scalable, extensible protocol stack that handles security, session establishment, and other aspects of mobile communications to make systems run more efficiently over today’s low-bandwidth wireless networks.
Instead of using HTML, WAP uses Wireless Markup Language (WML), a small subset of Extensible Markup Language (XML), to create and deliver content. As Figure 1 shows, the WAP gateway translates requests from the WAP protocol stack to the TCP/IP stack so they can be submitted to Web servers. The gateway translates WAP content into compact encoded formats that reduce the amount of data it sends over the low-bandwidth wireless network.

WAP is currently the most widely adopted wireless protocol in the world among carriers and handset manufacturers.

Launched in 1997 by Phone.com (now OpenWave), Motorola, Nokia, and Ericsson, the WAP Forum (http://www.wapforum.org) has grown to include more than 95 percent of the global handset market. The forum is planning to update the WAP protocol to ensure compatibility with 2.5G and 3G wireless standards. On 30 January 2001, OpenWave unveiled its product architecture for wireless general packet radio service and 3G systems. This new platform, which simplifies migration from current WAP-based mobile services deployed on 2G systems, supports not only WML, but also XHTML, HDML (Handheld Device Markup Language), Compact HTML (a subset of HTML that emphasizes text and simple graphics), and WAP 2.0, scheduled for release later this year. The updated protocol will support full-color graphics, multimedia, and, for wireless operators, subscriber management capabilities.

**iMode.** WAP is not the only protocol aimed specifically at the wireless Internet. In fact, in recent months, some have questioned WAP’s long-term viability, particularly in view of the explosive growth in competing technologies. The best-known non-WAP solution is iMode, a wireless Internet service that NTT DoCoMo introduced in Japan in February 1999. iMode relies on modifications and extensions of existing protocols. With iMode, smart-phone users can browse the Net with a touch of a button. iMode has a transmission speed of 9.6 Kbps, utilizes a packet-switched connection, and has adopted CHTML as its markup language.

**Other technologies**

Location-based services that use device location information to modify communications content are likely to become important for applications such as commerce and emergency services. Clearly, location-aware systems also raise complex privacy issues. Voice-based access is also likely to be important because of limited device display capabilities and because voice interaction has advantages in situations where keypad entry is impractical—for example, while driving a car. Indoor wireless networks are also likely to be extremely important for Internet access. Because wireless LANs and PANs have less severe bandwidth constraints than wide-area cellular networks, they allow access that is closer to “wired” Internet access.

**MIDDLEWARE CHALLENGES**

Common to WAP, iMode, and other similar solutions is the need to specifically recode Internet content for wireless devices. A content provider offering a Web site to both desktop and wireless users currently must maintain two parallel versions of the site customized to wired and wireless devices. Middleware, one of the most dynamic and yet least understood technologies, offers an alternative to manually replicating content. Its basic purpose is to seamlessly and transparently translate a Web site’s existing content to mobile devices that support numerous operating systems, markup languages, microbrowsers, and protocols. Creating a wireless presence using middleware, however, presents several key challenges.

**Application integration**

One challenge is integrating disparate content sources. Although some Web developers are beginning to store information in XML, most existing content was developed for desktop-based, nonmobile HTML browsers. Typically, most Web sites have a layered structure that closely ties a presentation layer to an underlying logic layer. However, wireless devices require a drastically different presentation layer, and rearchitecting the entire site to decouple these layers would be extremely expensive.

**Device independence**

Another challenge is the proliferation of devices, browsers, and markup languages. For example, markup languages include such variants as HDML.
Different browser features—support for non-nested tables and images, nested tables but no images, images and nested tables but only one font size, and so on—compound the problem. Display capabilities range from two-line black-and-white displays to full-color displays with tens of thousands of pixels.

Optimal user interface

Creating a compelling user interface that is appropriate for different device classes is another challenge. For example, a stock-trading site might want to expose a market research function containing charts and graphs to a PalmPilot but not to a limited-display cell phone. To avoid forcing the cell phone user to scroll down numerous lines or navigate through multiple menus to access desired content, the two devices’ information architecture would have to be drastically different.

**Middleware Solutions**

At first glance, it seems that the easiest way to address these problems is to rewrite existing Web content in a language appropriate for a particular protocol (see the “Internet Markup Languages” sidebar). This would involve, for example, creating WML-
formatted content to fit various WAP devices, CHTML-formatted content for iMode devices, and so on. However, the wide range of devices, browser functions, gateway interfaces, and markup language nuances make this approach intractable.

**User-transparent transformation**

In user-transparent transformation, a middleware application transparently reformats content “on the fly” into a user-specific presentation, interface, and protocol. In this context, a middleware application sits between the existing content server and the user agent. Because the middleware application can automatically detect the kind of device being used and format the contents accordingly, it is not necessary to maintain Web content in multiple formats.

User-transparent middleware approaches can process transcoding with or without a priori information. They can either strip down the content from an under-described source such as HTML or build up content from discrete pieces. However, end users still must scroll through hundreds of lines to get to the content in which they are interested.

**Relational Markup Language**

Relational Markup Language incorporates two concepts—relational hierarchies of content and adding context to content—to resolve these problems. Because it describes content through relationships, RML provides a write-once-deliver-anywhere solution. An interpreter can use the relational information to create an optimal information architecture. As Figure 2 illustrates, converting application content to a normalized RML format facilitates creating a modular architecture. When a new device, browser, gateway, or markup language emerges, adding one new module supports both current and future users.

**Atoms and groups.** The RML framework uses *atomics*—discrete pieces such as a word, sentence, paragraph, image, link, or any other piece of content—to create a framework that supports the addition of both relational data and context. RML encapsulates atomics into *groups* to create a complex relational hierarchy. As Figure 3 shows, groups contain one or more

---

**Figure 2.** Creating a modular architecture using Relational Markup Language. (a) A traditional wireless implementation can require different transcoding steps for each possible combination of markup languages. (b) In contrast, an RML-based implementation provides an intermediate format for the automatic markup of all markup languages. The output can then be generated without regard to the original input markup language.

**Figure 3.** Breaking Web content into atomics and groups. This typical news site’s content decomposes into discrete pieces—links, sentences, paragraphs, and images—to create a framework supporting the addition of both relational information and context. The main group consists of all the atomics and groups that make up the site. Each top story’s image, headline, and introductory text form another group. The headline group includes three subsections: world, business, and science. Finally, each subsection and its associated story links form a group.

---

**Budget Cuts Could Clip NASA’s Wings**

New US astronauts eager to “kick the tires and light the fires” may find that their best chance to soar, at least in the near future, comes when they board airliners for public-speaking tours. That is one implication of the new spending plan the White House is proposing for the US space program.

**World**

- Pakistani City Under Security Siege to Stop Rally
- Indonesia’s Wahid Rejects Second Censure Paper
- Israeli Killed in West Bank Shooting

**Business**

- Vodafone to Buy Stake in Japan Telecom-FT
- Marriott to Make Strong Meridien Bid
- Reliant Resources IPO Gets $1.56 Billion
- Pier 1 Imports Cuts Forecasts

**Science**

- Mirror Tests Reflect Dolphins’ Intelligence
- Peptide Fights Arthritis in Mice
- Phytopharm Looks for Cures in Herbs And Spices
atomics, and they can themselves be part of larger groups.

**Context.** An interpreter can add contextual attributes to the atomics and groups to optimize the information architecture for a specific mobile device. Adding context allows the RML document to handle current and future application calls that different devices make from the browser. The context attribute defines the type of application call, and context data defines the parameters the document needs to make the function call. Figure 4 shows how an interpreter can use relational information to create an optimal user experience.

RML also can differentiate between sequential and nonsequential content. For example, a page of text is sequential content, and a list of links to news stories is nonsequential content. If nonsequential content does not fit on one page—for example, the headlines subsection—RML collapses the associated groups into links.

Device-specific tasks may include click-to-dial, creating an address book entry, or sending e-mail. Developers can use the RML framework to develop algorithms that optimize the user’s experience on a variety of mobile devices.

**References**


**Subbas C. Saha** is a former manager of the research and technology group at 2Roam. His research interests include multimedia compression and communication, wireless Internet and networking, and soft computing.

**Mark Jamtgaard** is chief technical officer of 2Roam. His research interests include wireless communications and adaptive systems. He received a BS in electrical engineering from Santa Clara University. Contact him at mark@2roam.com.

**John Villasenor** is professor and vice chair of the Electrical Engineering Department at the University of California, Los Angeles. His research interests include communications, image and video coding, and configurable computing. Villasenor received a PhD in electrical engineering from Stanford University. Contact him at villa@icsl.ucla.edu.