

Editor: Sumi Helal ■ University of Florida ■ helal@cise.ufl.edu

# WAP: Present and Future

*Vijay Kumar, Srinivas Parimi, and Dharma P. Agrawal*

In 1997, several wireless-phone manufacturers organized an industry group called the Wireless Application Protocol forum. This group defined the WAP specification in the form of a long, describes technical document series that defines standards for implementing wireless-network applications. Hundreds of industries strongly supported the WAP forum for standardization to help the technology become widely adopted.

Unfortunately, most of the enthusiasm surrounding WAP has evaporated owing to inherently low channel bandwidth, increased round-trip delays, and a lack of security. WAP technology is primitive and still evolving, and its future depends on how quickly it can improve the transfer rate and effectively enhance its business model.

## THE WAP ARCHITECTURE

In the last few years, WAP has emerged as a standard Internet-enabling wireless protocol and a browser framework for small, limited-display-capable devices. WAP allows Internet access to cell phones, PDAs, and other low-computational-power devices. In addition, the lightweight WAP protocol has a layered architecture and is designed to operate over a variety of wireless services, including Code Division Multiple Access, Cellular Digital Packet Data, General Packet Radio Service (GPRS), and so on.[1] (For more information on the layered architecture, see the "WAP Layers" sidebar.)

Based on Internet standards such as HTTP, WAP wireless protocols[2–4] require transferring large amounts of mainly text-based data. The WAP standard consists of two essential elements: an end-to-end application protocol and an application environment based on the browser. The application protocol is a communication stack embedded in each WAP-enabled wireless device (also known as the *user agent*). The server side defined as a WAP *gateway* implements the other end of the protocol, which can communicate with any WAP *client*.

The WAP network structure (see Figure 1[5]) sets up a session using these steps:[6,7]

1. A mobile telephone sends WAP requests to a WAP gateway.
2. The gateway, upon receiving a WAP request, sends an HTTP request to a plain Web server, which provides the content through a normal HTTP response (the Web server perceives the gateway as a proxy server).
3. The gateway converts the HTTP response into a WAP response for the mobile device.
4. The microbrowser in the mobile terminal interprets the response and displays it appropriately.

## WAP SUPPORTING COMPONENTS

WAP technology has three major supporting components. The first is the Wireless Markup Language. WML is the WAP equivalent to HTML and is based on XML.[8] It makes optimal use of small screens, with a built-in scalability from two-line text displays to the full graphic screens on smart phones and communication devices. The Wireless Application Environment specification defines the syntax, variables, and elements used in a valid WML file. WML employs the concept of *decks* and *cards*. Each card is a frame displayed on the screen. We refer to a logical collection of interlinked cards as a deck, usually stored in a single WML file.

The second is WMLScript, a client-side scripting language used with WML that makes WML pages dynamic (similar to what JavaScript[9] allows with HTML). WMLScript makes minimal demands on memory and CPU usage, omitting a number of functions that are not required and that are present in other scripting languages for wireless applications. Regular scripting languages are resource intensive and can't be used to display interactive pages on a mobile phone. WMLScript is a scaled-down, simplified script, developed specifically to fit the WAP architecture of decks and cards. It lets the developer provide interactivity in WAP pages without taxing the valuable wireless resources.

The third supporting component is Wireless Bitmaps. WBMP is the default picture format for WAP. WBMPs are uncompressed, monochrome black-and-white bitmaps for use in devices with small screens and narrow bandwidth connections (see Figure 2). However, the screen size and bandwidth,

1536-1268/03/$17.00 © 2003 IEEE ■ *Published by the IEEE CS and IEEE Communications Society*

STANDARDS, TOOLS & BEST PRACTICES

## WAP LAYERS

The WAP protocol stack has five layers: application, session, transaction, security, and transport (see Figure A[1]). Each layer performs almost the same functions as the corresponding layers of the Internet model.

### Application

The application layer consists of the Wireless Application Environment and user agents. The most common type of user agent in the WAP architecture is a browser meant to interpret Wireless Markup Language and WMLScript. User agents that endeavor to provide services beyond those of a browser generally take advantage of the WAP specification's Wireless Telephony Application features.

### Session

The Wireless Session Protocol presents the application layer with a way to uniformly receive both "reliable" connection-oriented and "unreliable" connectionless transmissions. The WSP facilitates transmission mechanisms such as

- Providing HTTP functionality
- Enabling users to participate in long-lived data transmission sessions
- Allowing a server application to determine whether a client can support certain protocol facilities and configurations (this is known as "a capability negotiation")

### Transaction

The Wireless Transaction Protocol handles requests and responses to and from the user agent to the application server. The WTP concentrates on transaction services for online activities such as Web browsing. It is designed to decrease the number of transaction phases that typical wired-oriented protocols require.

### Security

The Wireless Transport Layer Security implements many features to ensure secure data transmissions and to protect the users, the network and service operators, and the functionality of the upper layers of the WAP stack. The WTLS provides safe data transmission that could support services requiring a high level of security such as mobile e-commerce.

### Transport

The Wireless Datagram Protocol transmits and receives data to and from the user agents. The WDP can exchange information with many types of wireless data carrier technologies or "bearers." It effectively hides the differences in bearer technologies from the rest of the stack and provides a common interface to the upper-layer protocols. Layers can function independently of the underlying network and the wireless devices.

#### REFERENCE

1. D.P. Agrawal and Q.-A. Zeng, *Introduction to Wireless and Mobile Systems*, Brooks/Cole Publishing, Pacific Grove, Calif., 2003.

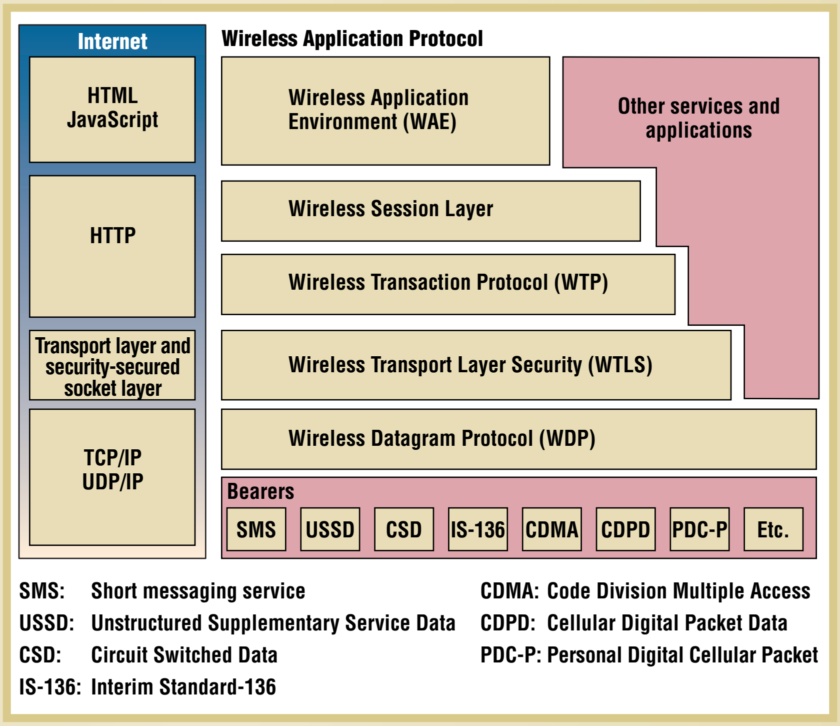


**Figure A. The five Wireless Application Protocol layers.[1]**

along with limited graphics capabilities, are WBMP's major constraints. Recent developments in display technology provide color bitmaps for the latest WAP browsers.

## WAP ADVANTAGES

Many inherent WAP characteristics offer major advantages. First, WAP saves time and money. A WAP-based application's ability to send and receive data in real time lets companies make field changes and coordinate their staff more efficiently. Companies can also speed up and automate their distribution cycle, thus minimizing the required manpower.

Second, there's no hardware obsolescence. One of the greatest concerns for companies with mobile workers is the huge investment in handheld hardware. Companies would be interested if the standard could work on legacy hardware. This would help them increase the lifetime of their prior investments in mobile devices.

Finally, WAP allows multiplatform functionality. Because WML is based largely on XML,[8] little program modification is required to run WAP-based applications on traditional Web sites.

## IMPORTANT WAP APPLICATIONS

Given that a continuously growing percentage of commerce is happening over the Internet and that the number of mobile phone subscribers is increasing, the potential for growth of mobile commerce and hence WAP applications is unparalleled.

WAP provides an open technology platform for offering new, innovative services to the consumer market and a wireless channel for existing services. Examples of some such applications are financial trading, buying tickets online, ordering from restaurants, updating financial portfolios, conducting banking transactions (such as transferring funds between accounts), and comparison shopping.

WAP has also led to sales force and field service automation. Mobile access to corporate Intranets lets employees and business partners access data in a cost- and time-efficient manner. WAP has let companies move one step closer to real-time operations for faster, better decision-making and enhanced business performance.

In addition, WAP's invention has led to the development of *WAP telephony applications* that add a telephony element to the basic WAP infrastructure and make existing wireless network capabilities available to enhance WAP services. WTAs provide a framework for integrating wireless data and voice capabilities to create innovative services.

## WAP CRITIQUES

Just as there is no dearth of WAP supporters, the number of WAP bashers is growing. WAP technology is not flawless, and criticism from leading technologists fall into the following categories.

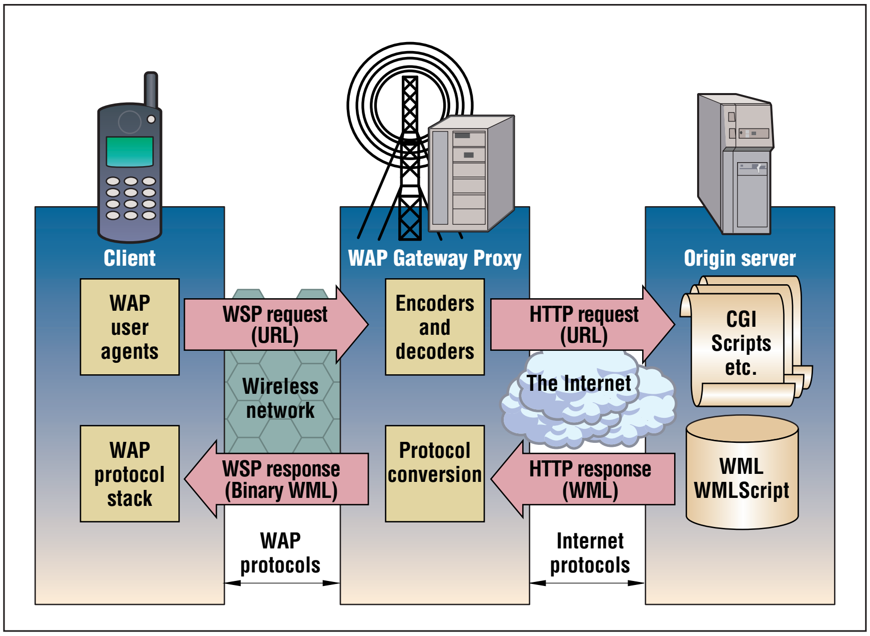


Figure 1. A Wireless Application Protocol network architecture.[5]

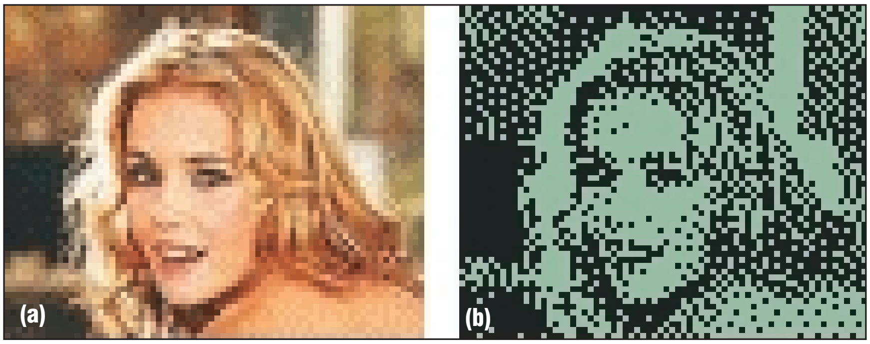


Figure 2. Two images: (a) normal and (b) Wireless Bitmap.

### WAP as a band-aid

WAP critics say that WAP is only a temporary fix. We will need it only until greater bandwidth in the communication channel and increased functionality in the personal terminal are widely available. GPRS-enabled PDAs, which deliver 28.8 kbps, are already hitting the European market. Carriers such as Verizon and Sprint are building 3G networks capable of handling data and video.

### Wrong bearer services

WAP is expensive when used with current bearer technologies. With circuit-switched data, we can expect long connection calls for any application that uses interactivity with the Internet or that lets the user select more information.

### Weak link

The WAP gateway must decrypt secure data coming from a caller using Wireless Transport Layer Security, and then it must re-encrypt the data before forwarding it to the bank or credit card company (or anyone else receiving it). The reverse must also be maintained. There is a window of vulnerability when the data is momentarily unencrypted.

STANDARDS, TOOLS & BEST PRACTICES

### WAP designer's nightmare

One of the biggest problems WAP developers face is that their applications will likely be rendered in many different ways on different mobile phones. Old WAP browsers support HDML (Handheld Device Markup Language); later browsers support WML. Different browsers treat certain tags and commands in different ways, and myriad devices with the same browser also function unpredictably. Generally, this lack of coordination among handset providers, browsers, emulators, and the WAP standard creates a frustrating environment for even the best developers.

### User experience

So far, WAP has not lived up to its promise of being an exciting way to access the Internet. Tradeoffs between functionality, performance, and design have dampened user experiences. WAP applications with broken links seem to be the norm, not the exception. Some people have difficulty just figuring out how to access the microbrowser on their mobile handset. Furthermore, after waiting for 30 to 45 seconds to connect, users find what looks like a walled garden that allows access only to certain sites. Access to options such as "go to other sites" is cumbersome and frustrating. Inputting a valid WAP URL through the existing numeric keyboard is tedious.

### Killer applications

The hype involving WAP has centered on the killer applications, which were touted to herald a new era of mobile commerce. However, even after almost five years of WAP development, a stable killer application has yet to see the light of day. Many have not yet been developed or, if developed, have not been marketed well enough to catch consumers' attention.

### Pricing

The pricing of WAP services has been problematic for both developers and consumers. On the developer side, operators have not shared the wealth with content providers and have essentially made it difficult for content providers to get any sort of valid return for their efforts. No true, measurable incentives exist to develop WAP content in the absence of wireless advertising and other revenue agreements.

Many carriers offer an expensive flat-rate pricing fee to consumers, in addition to subtracting voice minutes for time spent on data. This impacts WAP services in two ways. First, carriers have no efficient mechanism to establish fair revenue sharing with content providers. Second, consumers are historically price-sensitive in the services industry and view WAP as too expensive. An example of how a correct business model can help WAP is Telesim, Turkey's second-largest GSM operator. It eliminated its monthly fee for WAP usage and experienced a tenfold growth in its subscribers.

## THIN VERSUS FAT CLIENTS

WAP, also implemented as a *thin client*, uses a centralized server to store, fetch, and send data. The meager storage and low-memory capacity would be sufficient to run varied applications using a thin client. Remote-server-handling capability depends on the bandwidth of the communication media between the server and the client. On the other hand, *fat clients* implemented by Java 2 Micro Edition and the .NET Compact Framework make storage and processing more local than done remotely (for more information on J2ME, see the Jan.–Mar. and Apr.–June 2002 installments of this department; for information on .NET-CF, see the July–Sept. 2002 installment). Fat clients require large memory and storage capacities.

Each client type has its advantages and disadvantages. However, it is difficult to say which is better, because both are application oriented and the technology is constrained by technological limitations. Whether wireless messaging should employ a thin or fat client is still debatable.

The obvious disadvantage of a thin client might be its limited feature set, which could worsen user experiences. Fat clients face the storage constraints and limited processing power of mobile devices. Applications such as the Virtual Private Network (www.vpnc.org) and Virtual Network Connectivity (www.uk.research.att.com/vnc) have allowed virtual access to desktops from any computer. These technologies rely on the large bandwidth of the wired medium. Much research has shown that thin-client technology for desktops might not be a good practice for PC management.[10]

The latest developments in memory and storage technologies for handheld devices are enabling fat-client architecture by reducing the burden on the bandwidth, because local processing would eliminate some of the data transfer done at the remote server. Because mobile devices go wherever we go, WAP technology makes them suitable for the current storage, memory, and processing capabilities of mobile devices. Unless there are astronomical advances in the wireless medium to allow larger bandwidths, fat clients seem to be the way to go for future handheld devices.

WAP optimists have been quick to rebut WAP shortcomings. WAP security has been enhanced in the new version 2.0, and most shortcomings have more to do with an inadequate business model than the WAP technology itself. Also, analysts suggest that if WAP is incompatible with current technologies, GPRS might make it more practical.

WAP is an application protocol suite designed to function over any bearer service. This is where GPRS comes into practice as an ideal bearer for WAP resource requirements; its higher transfer rates are needed to accommodate WAP's unusually complex and frequent network resource requirements. Furthermore, the increased bandwidth will allow better and more varied applications with pictures, video, and sound. Japan and South Korea are already carrying packet data on their networks, and

Europe is adhering to the GSM standard with many confirmed GPRS contracts.

WAP technology could bring about the convergence of mobile communications and the Internet. With any new technology come risks, bugs, and opportunities. With WAP, the opportunity and adventure definitely outweigh the risks. However, WAP has not lived up to its lofty expectations, and its growth has taken a beating. In responding to the recent backlash against WAP, advocates quickly point out that WAP exists today in a very primitive form. Still, with the introduction of much faster bearer services such as GPRS or UMTS and a proper consumer-oriented business model, the struggling WAP might just get a much-needed lifeline.

## ACKNOWLEDGMENTS

We thank Yamini K. Patri of West Virginia University for providing useful information.

## REFERENCES

1. D.P. Agrawal and Q.-A. Zeng, *Introduction to Wireless and Mobile Systems*, Brooks/Cole Publishing, Pacific Grove, Calif., 2003.
2. *Wireless Application Protocol (WAP)*, Int'l Eng. Consortium, www.iec.org/tutorials/acrobat/wap.pdf.
3. *WAP: An Introduction to the Wireless Application Protocol*, Mobile Lifestreams, 1999, www.totaltele.com/whitepaper/docs/Moblifstrms_WAP.pdf.
4. *Wireless Application Protocol: The Corporate Perspective*, white paper, Nokia, 1999, www.nokia.com/corporate/wap/files/whitepaper.pdf.
5. *Success 4 WAP*, white paper, Cisco Systems, 2001, www.mobilewhitepapers.com/pdf/wap.pdf.
6. Yi-Bing Lin and Imrich Chlamtac, *Wireless and Mobile Network Architectures*, John Wiley & Sons, New York, 2001.
7. *WAP White Paper*, Mobileinfo.com, www.mobileinfo.com/WAP/model.htm.
8. D. Hougland and K. Zafar, *Essential WAP for Web Professionals*, Prentice Hall, Upper Saddle River, N.J., 2001.
9. D. Flanagan, *JavaScript: The Definitive Guide*, 3rd ed., O'Reilly & Associates, Sebastopol, Calif., 2001.
10. P. Lowber, "Thin Client vs. Fat Client TCO," research note, Gartner, 2001, www.compaq.co.uk/lifecycle/reports/docs/thinclient.pdf.

**Vijay Kumar** is a graduate student in computer engineering at the University of Cincinnati. He has joined the wireless research and development center of Texas Instruments in Berlin, Germany. His research interests include second- and third-generation protocols and mobile and wireless ad hoc networks. He received his BE in electronics and communication from the Regional Engineering College, Allahabad, India. Contact him at k_vijay@hotmail.com.

**Srinivas Parimi** is pursuing his doctoral degree in electrical engineering, specializing in MEMS, at the University of Cincinnati. He is also interested in computer networks and automation. He received his master's in electrical engineering from the University of Cincinnati. Contact him at sparimi@ececs.uc.edu.

**Dharma P. Agrawal** is the Ohio Board of Regents Distinguished Professor of Computer Science and Computer Engineering at the University of Cincinnati. He is the founding director of the Research Center for Distributed and Mobile Computing. His research interests include energy-efficient routing and information retrieval in ad hoc and sensor networks, effective handoff and multicasting in integrated wireless networks, interference analysis in piconets and routing in a scatternet, use of directional antennas for enhanced quality of service, and scheduling of periodic real-time applications. He received his DSc from the Federal Institute of Technology, Lausanne, Switzerland. Contact him at dpa@ececs.uc.edu.