The Mobile Internet and Wireless Web

Traditionally, the Internet has been accessed via a 2 Mbps cable modem from the office or home, and the content has been received on high-resolution screens via hypertext transfer protocol (HTTP) in hypertext markup language (HTML) format. But with the advent of and considerable interest in small portable devices and handhelds, capable of accessing the Web via wireless links like GPRS or GSM, focus has shifted to tiny displays of 96 x 64 pixels to receive content 24 x 7 x 365.

The wireless application protocol (WAP) is an open, global specification that empowers mobile users with wireless devices to easily access e-mail and the Web, through information and services instantly. The WAP Forum was initially set up as a consortium led by Nokia, Ericsson and Motorola in 1997, as an Industry Group for the purpose of extending the existing Internet standards for use with wireless communication. It was able to deliver services as early as 1999, through the WAP standard 1.1, providing speeds of 9.6 Kbps. The WAP standard 2.0 came in 2002 and could provide speeds of 384 Kbps. As of 2002, more than 500 companies from all parts of the industry, including network operators, device manufacturers, service providers and software vendors, are members of the WAP Forum.

In this chapter, we concentrate on the WAP and its gateway, but before we go into the details of WAP, let us take a look at the traditional Web programming model and see how it compares with the WAP programming model

9.1 The Web programming model

The Internet World-Wide-Web (WWW) architecture provides a very flexible and powerful programming model, shown in Figure 9.1.

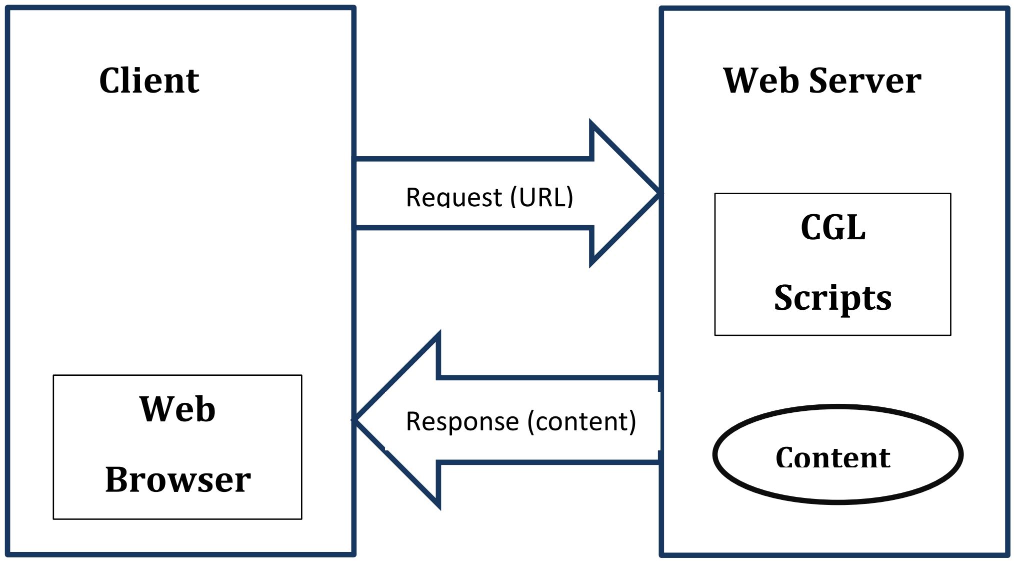

Figure 9.1 The Web Programming Model

In this model, applications and content are presented in standard data formats and are browsed by applications known as Web browsers. The Web browser is a networked application; that is; it sends requests for named data objects to a network server, and the network server responds with the data encoded using the standard formats. The WWW standards specify many of the mechanisms necessary to build a general-purpose application environment, including the following:

1. Standard naming model: All servers and content on the WWW are named with an Internet standard **uniform resource locator** (URL).

2. Content typing: All content on the WWW is given a specific type, thereby allowing Web browsers to correctly process the content, based on its type.

3. Standard content formats: All Web browsers support a set of standard content formats. These include the HTML, scripting languages like ECMAScript and Javascript, and a large number of other formats.

4. Standard protocols: Standard networking protocols allow any Web browser to communicate with any Web server. The most commonly used protocol on the WWW is the HTTP, operating on top of the TCP/IP protocol suite.

This infrastructure allows users to easily reach a large number of third-party applications and content services. It also allows application developers to easily create applications and content services for a large community of clients.

9.2 The WAP programming model

The WAP programming model, shown in Figure 9.2, is similar to the Web programming model, with certain enhancements and extensions to match the characteristics and constraints of the wireless environment. These are related to slow central processing units (CPUs), low-bandwidth connections, small memory and small screens. Wherever possible, existing standards have been adopted or have been used as the starting point for the WAP technology, as this provides flexibility to application developers. The use of proxies is one such feature in this direction.

The WWW programming model provides several benefits, including a familiar programming model, a proven architecture and use of existing tools like Web servers, extensible markup language (XML) tools, etc. Optimizations and extensions have been made in order to match the characteristics of the wireless environment. The most significant enhancements that WAP has added to the programming model are push mechanism and support for telephony (WTA).

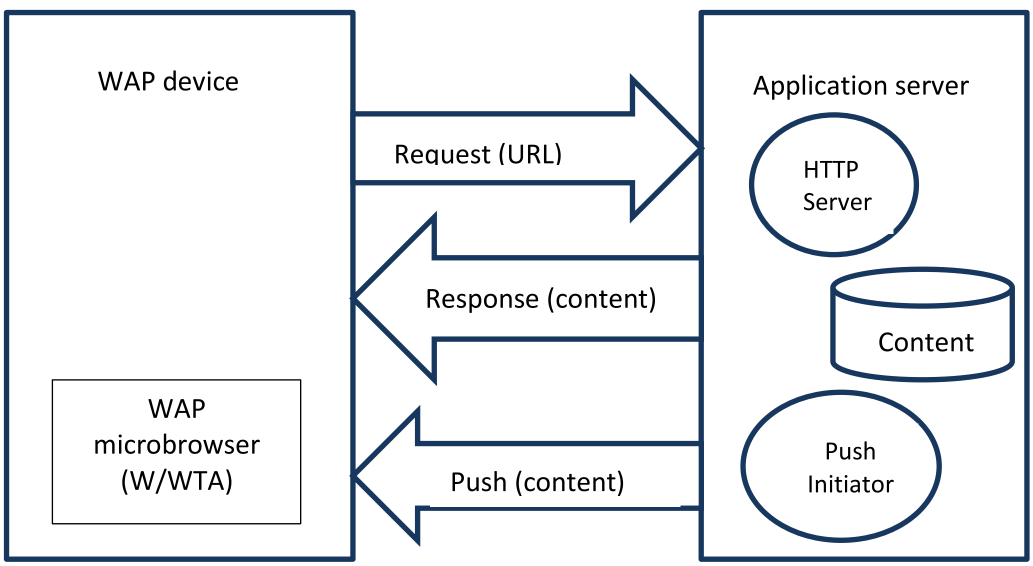

Figure 9.2 The WAP Programming Model

The classical request-response mechanism is commonly referred to as **pull**, where the client pulls data from the server, in contrast with the **push** mechanism, where the server pushes data to the client. WAP content and applications are specified in a set of well-known content formats, based on the familiar WWW content formats. Content is transported using a set of standard communication protocols based on the WWW communication protocols. The WAP microbrowser in the wireless terminal coordinates the user interface and is analogous to a standard Web browser. WAP defines a set of standard components that enable communication between mobile terminals and network servers, including the following:

1. Standard naming model: WWW standard URLs are used to identify WAP content on origin servers. WWW standard **uniform resource indicators** (URIs) are used to identify local resources in a device, for example, call control functions.

2. Content typing: All WAP content is given a specific type consistent with WWW typing, thereby allowing WAP user agents to correctly process the content, based on its type.

3. Standard content formats: All WAP content formats are based on WWW technology and include display markup, calendar information, electronic business card objects, images and scripting languages, and a large number of formats.

4. Standard protocols: WAP communication protocols enable the communication of browser requests from the mobile terminal to the Web server. The WAP content types and protocols have been optimized for mass market, handheld wireless devices.

9.3 WAP protocol stack

The WAP 1.1 protocol stack is shown in Figure 9.3. It is derived from the ISO OSI reference model and has five different layers.

The application layer (WAE) provides an application environment for the development and execution of portable application and services. The microbrowser sits here but does not use HTML. Instead, it uses the wireless markup language (WML), which is an application of XML.

Therefore, a WAP device can only access those pages that have been converted to WML. However, it is possible to design an on-the-fly HTML to WML filter to increase the set of pages available. This is called the WAP gateway and is the subject of discussion in Section 9.5.

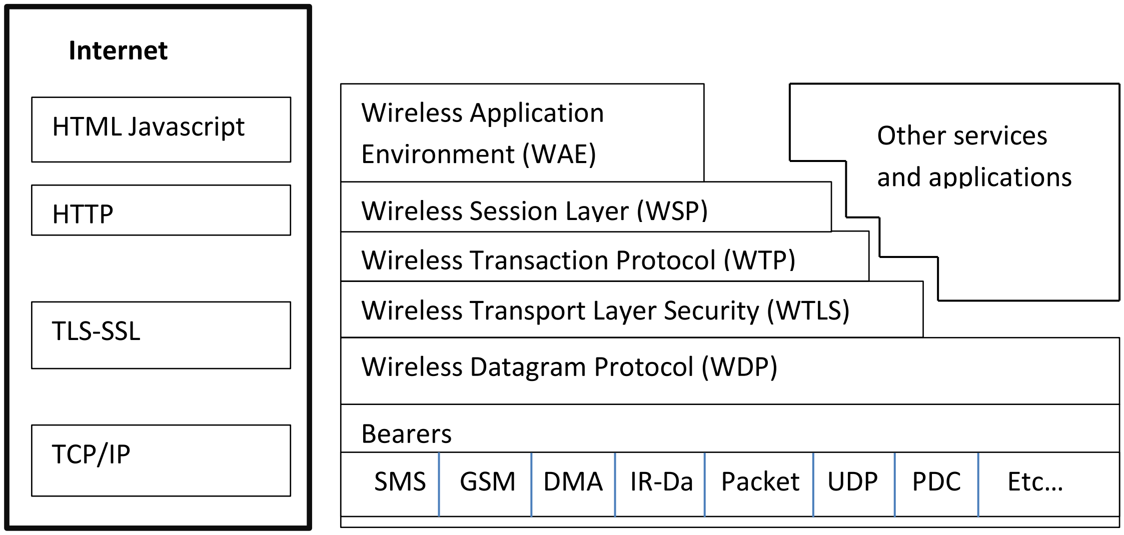

Figure 9.3 WAP 1.1 Protocol Stack

The session layer (WSP) supplies methods for the organized exchange of content in client/server applications. It is similar to HTTP/1.1, but with some restrictions and extensions for optimization purposes.

The transaction layer (WTP) provides different methods for performing transactions, to a varying degree of reliability. It manages requests and responses, either reliably or unreliably.

The security layer (WTLS) is an optional layer that provides, when present, authentication, privacy and secure connections between applications. WTLS is a subset of Netscape's Secure Socket Layer (SSL).

The transport layer (WDP) is the bottom layer of the WAP stack, which shelters the upper layers from bearer services offered by the operator. It is essentially user datagram protocol (UDP).

The bearer layer supports all the existing mobile phone systems, including GSM, DAMPS and CDMA. The WAP 1.1 data rate is 9,600 Bps.

WAP 1.1 is basically a circuit-switched system that can be used with a variety of different networks. But it comes with a high per-minute charge. Its other major drawback is that it does not support HTML, and hence content available to the WAP device is limited. Another major drawback with WAP 1.1 is that it does not support Internet standard protocols. Hence, a gateway is required to translate contents from the WAP environment to the Internet environment. This poses a major security risk, which we will discuss in Chapter 11.

9.4 Information-mode (l-mode)

In 1999, the Japanese NTT DoCoMo company launched the information-mode (i-mode) for wireless access to the Web, using special handsets. The i-mode system has a new transmission system, a new handset and a new language. The new transmission system is built on two separate networks- the existing circuit-switched mobile phone network for voice and a new packet-switched network built specifically for i-mode service. These are, however, not usable concurrently.

I-mode handsets are basically wireless terminals and are not user programmable. When the i-mode handset is switched on, the user is presented with a list of services categorized as e-mail, news, weather, sports, games, shopping, etc. Each of these services is to be subscribed and paid for, though nominally.

The language used in the i-mode is a compact version or subset of HTML, called cHTML. While most of the HTML tags are allowed here, the cHTML browser does not support Javascript, frames, style sheets. JPEG images, etc. However, the l-mode server is a full-blown system supporting CGI, Perl, PHP, JSP, ASP, etc.

9.5 WAP 2.0

The second-generation wireless Web, WAP 2.0, uses a packet-switching mechanism like GPRS. It has some new features, the most significant of which are as follows:

1. Push as well as pull model.
2. Support for integrating telephony into applications.
3. Multimedia messaging: WAP 2.0 supports the merger of voice and data in a variety of ways. Along with e-mail and telephony, multimedia messaging is supported.
4. Inclusion of 264 pictograms: Pictograms are included in a number of categories, namely, animals, appliances, dress, emotion, food, human body, gender, maps music, plants, etc.
5. Interface to a storage device: Flash ROM is supported as a storage device. A WAP-enabled wireless camera could use it for temporary image storage, before sending it on the Internet.
6. Support for plug-ins in the browser: Plug-ins extend the browser's capabilities, with provision for scripting languages, etc.

Various technical differences are also present between WAP 1.1 and 2.0. Firstly, WAP 2.0 continues to support the WAP 1.1 stack, but also supports the standard Internet stack with TCP and HTTP/1.1. However, four minor but compatible changes to TCP were made to simplify the code:

1. Use of a fixed 64-KB window
2. No slow start
3. Maximum MTU of 1500 bytes, and
4. A slightly different retransmission algorithm

Secondly, WAP 2.0 supports XHTML Basic, a markup language intended for small wireless devices like mobile phones, televisions, personal digital assistants (PDAs), vending machines, pagers, cars, game machines, watches, etc. It therefore does not support style sheets, scripts or frames, but most of the standard tags are there, which are defined in XML.

WAP 2.0 runs at 384 Kbps, which is still very slow, as compared to the 11 Mbps or 54 Mbps data rates offered by IEEE Standard 802.11, which is giving it huge competition.

9.6 WAP gateway

A gateway is an intermediary element, usually used to connect two different types of networks. It receives requests directly from the clients as if it were an origin server that the clients want to retrieve the information from. Clients are usually not aware that they are speaking to the gateway. This scenario is shown in Figure 9.4.

A WAP gateway acts as a bridge between the Internet world and the mobile world, by providing access to and translation of the Internet resources. The architecture of the protocol used at the WAP gateway is shown in Figure 9.5.

WAP uses coded binary data to improve transmission efficiency. The header and content are compactly compiled. For the traditional HTTP network, the packet header is in string format. In order to adapt to the WAP network, encoding and decoding techniques are required. The coder/decoder functionality within the gateway is used to convert the WML and WMLScript content going to and coming from the client into a form that is optimized for low-bandwidth networks. Figure 9.6 illustrates the functionality of the WAP gateway.

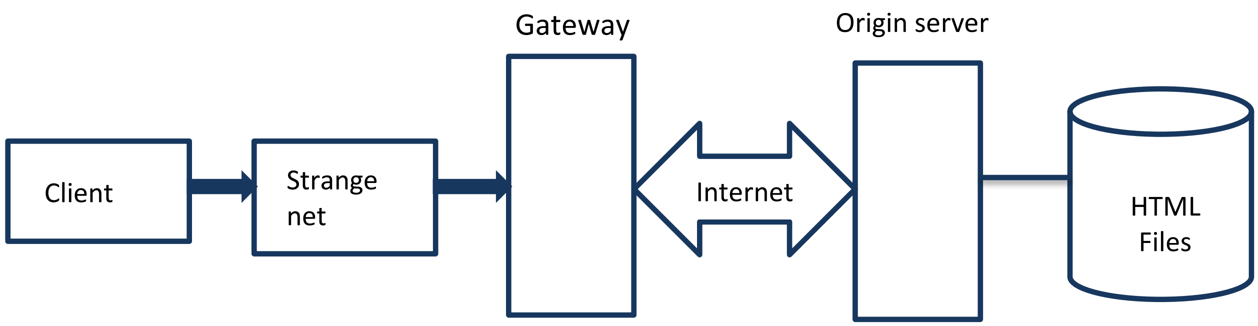

Figure 9.4 A Gateway

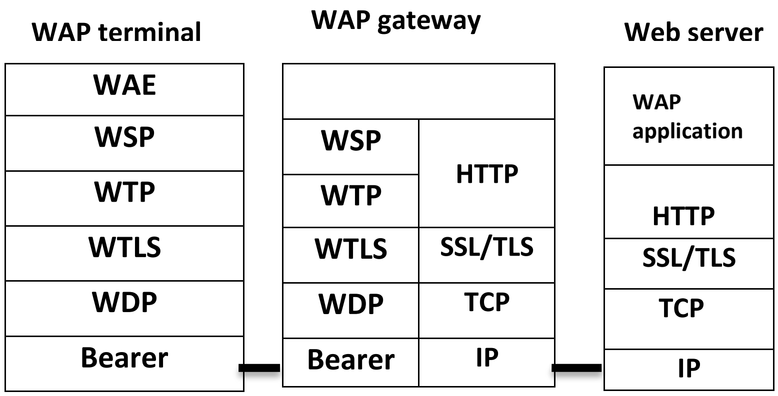

Figure 9.5 Architecture of the WAP Gateway Protocol

To translate the traditional HTTP requests and replies to the coded forms, a mapping table is required. The features provided by the WAP gateway are pull operation, push operation, on-the-fly image conversion (GIF, JPEG to wireless bit map protocol [WBMP]), WML to WAP binary XML (WBXML) (compressed format), access control, logging and cache service.

A WAP gateway does three tasks:

1. Header translation, which allows the client system to access the Internet via a different protocol.
2. Push operation, which allows the server to send the right information to the client, and
3. Content compilation, which allows compaction of the data for low-bandwidth support.