WAP Architecture

Version 12-July-2001

Wireless Application Protocol
Architecture Specification
WAP-210-WAPArch-20010712

A list of errata and updates to this document is available from the WAP Forum ™ Web site, http://www.wapforum.org/in the form of SIN documents, which are subject to revision or removal without notice.

Copyright 2000-2001 Wireless Application Protocol Forum Ltd. All Rights Reserved. Terms and conditions of use are available from the Wireless Application Protocol Forum Ltd. Web site (http://www.wapforum.org/what/copyright.htm).

EX-1013
U.S. Patent 8,825,801

© 2001, Wireless Application Protocol Forum, Ltd. All rights reserved.

Terms and conditions of use are available from the WAP Forum™ Web site at http://www.wapforum.org/what/copyright.htm.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. You may not use this document in any other manner without the prior written permission of the WAP Forum™. The WAP Forum authorises you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services offered by you.

The WAP Forum™ assumes no responsibility for errors or omissions in this document. In no event shall the WAP Forum be liable for any special, indirect or consequential damages or any damages whatsoever arising out of or in connection with the use of this information.

WAP Forum™ members have agreed to use reasonable endeavors to disclose in a timely manner to the WAP Forum the existence of all intellectual property rights (IPR's) essential to the present document. The members do not have an obligation to conduct IPR searches. This information is publicly available to members and non-members of the WAP Forum and may be found on the "WAP IPR Declarations" list at http://www.wapforum.org/what/ipr.htm. Essential IPR is available for license on the basis set out in the schedule to the WAP Forum Application Form.

No representations or warranties (whether express or implied) are made by the WAP Forum™ or any WAP Forum member or its affiliates regarding any of the IPR's represented on this list, including but not limited to the accuracy, completeness, validity or relevance of the information or whether or not such rights are essential or non-essential.

This document is available online in PDF format at http://www.wapforum.org/.

Known problems associated with this document are published at http://www.wapforum.org/.

Comments regarding this document can be submitted to the WAP Forum™ in the manner published at http://www.wapforum.org/.

Document History	
WAP-210-WAPArch-20010712-a	Current

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

Contents

1. SCOPE **4**

2. DOCUMENT STATUS **5**

2.1 COPYRIGHT NOTICE 5
2.2 TRADEMARK NOTICE 5
2.3 ERRATA 5
2.4 COMMENTS 5

3. REFERENCES **6**

4. DEFINITIONS AND ABBREVIATIONS **8**

4.1 DEFINITIONS 8
4.2 ABBREVIATIONS 8

5. BACKGROUND **11**

5.1 MOTIVATION 11
5.2 ARCHITECTURAL GOALS 12

6. ARCHITECTURE OVERVIEW **12**

6.1 THE WORLD-WIDE WEB MODEL 12
6.2 THE WAP MODEL 13
6.3 FEATURE/PERFORMANCE-ENHANCING PROXIES 14
6.4 SUPPORTING SERVERS 15
6.5 WAP NETWORK ELEMENTS 16
6.6 DEVICE ARCHITECTURE 17
6.7 SECURITY MODEL 17

7. COMPONENTS OF THE WAP ARCHITECTURE **18**

7.1 BEARER NETWORKS 18
7.2 TRANSPORT SERVICES 19
7.3 TRANSFER SERVICES 19
7.4 SESSION SERVICES 19
7.5 APPLICATION FRAMEWORK 20
7.6 SECURITY SERVICES 20
7.7 SERVICE DISCOVERY 21
7.8 OTHER SERVICES AND APPLICATIONS 21

8. SAMPLE CONFIGURATIONS OF WAP TECHNOLOGY **21**

9. CONFORMANCE AND INTEROPERABILITY **24**

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

1. Scope

The Wireless Application Protocol (WAP™) is a result of continuous work to define an industry wide specification for developing applications that operate over wireless communication networks. The scope for the WAP Forum is to define a set of specifications to be used by service applications. The wireless market is growing very quickly, reaching new customers and providing new services. To enable operators and manufacturers to meet the challenges in advanced services, differentiation, and fast/flexible service creation, WAP selects and defines a set of open, extensible protocols and content formats as a basis for interoperable implementations.

The objectives of the WAP Forum are:

- To bring Internet content and advanced data services to digital cellular phones and other wireless terminals.
- To create a global wireless protocol specification that will work across differing wireless network technologies.
- To enable the creation of content and applications that scale across a very wide range of bearer networks and device types.
- To embrace and extend existing standards and technology wherever appropriate.

The WAP Architecture Specification is intended to present the system and protocol architectures essential to achieving the objectives of the WAP Forum. The WAP Architecture Specification acts as the starting point for understanding the WAP technologies and the resulting specifications. As such, it provides an overview of the different technologies and references the appropriate specifications for further details.

This version of the WAP Architecture continues the themes and builds on the successes of the initial WAP architecture. Network elements remain similar in function. For example, the architecture uses performance and feature-enhancing proxies to offload processing from constrained devices, to expose features and functions of the wireless network, and to provide for network and service management. This version of the architecture has been enhanced to allow for a broader selection of connection paths between clients and origin servers as necessary, for example to provide end-to-end security.

The WAP Architecture Specification itself provides a framework for a variety of protocols, features, and services. It does not mandate any specific implementation and shall therefore be considered informative.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

2. Document Status

This document is available online in the following formats:

- PDF format at http://www.wapforum.org/.

2.1 Copyright Notice

© Copyright Wireless Application Forum Ltd, 2000.

Terms and conditions of use are available from the Wireless Application Protocol Forum Ltd. web site at http://www.wapforum.org/docs/copyright.htm.

2.2 Trademark Notice

WAP and all WAP-based marks are world-wide trademarks or registered trademarks of Wireless Application Protocol Forum Ltd.

2.3 Errata

Known problems associated with this document are published at http://www.wapforum.org/.

2.4 Comments

Comments regarding this document can be submitted to the WAP Forum in the manner published at http://www.wapforum.org/.

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

3. References

[ClassConform] WAP Class Conformance Requirements, WAP Forum.

[ECMAScript] Standard ECMA-262: "ECMAScript Language Specification", ECMA, June 1997

[HTML4] "HTML 4.0 Specification, W3C Recommendation 18-December-1997, REC-HTML40-971218", D. Raggett, et al., September 17, 1997. URL: http://www.w3.org/TR/REC-html40

[HTTPState] “HTTP State Management”, WAP-223-HTTPSM, WAP Forum.

[JavaScript] "JavaScript: The Definitive Guide", David Flanagan. O’Reilly & Associates, Inc. 1997

[MMSEncapsulation] “WAP Multimedia Messaging Service Message Encapsulation”, WAP Forum.

[ProvArch] “WAP Provisioning Architecture Overview”, WAP Forum.

[PushArchOverview] “WAP Push Architectural Overview”, WAP Forum.

[PushOTA] “WAP Push OTA Protocol”, WAP Forum.

[RFC2045] "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", N. Freed, et al., November 1996. URL: http://www.rfc-editor.org/rfc/rfc2045.txt

[RFC2048] "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", N. Freed, et al., November 1996. URL: http://www.rfc-editor.org/rfc/rfc2048.txt

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997. URL: http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2246] “The TLS Protocol Version 1.0”, T. Dierks, C. Allen, January 1999. URL: http://www.rfc-editor.org/rfc/rfc2246.txt

[RFC2396] “Uniform Resource Identifiers (URI): Generic Syntax”, T. Berners-Lee, R. Fielding, L. Masinter. August 1998. URL: http://www.rfc-editor.org/rfc/rfc2396.txt

[RFC2401] “Security Architecture for the Internet Protocol”, S. Kent, R. Atkinson, November 1998. URL: http://www.rfc-editor.org/rfc/rfc2401.txt

[RFC2616] "Hypertext Transfer Protocol - HTTP/1.1", R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999. URL: http://www.rfc-editor.org/rfc/rfc2616.txt

[RFC2617] “HTTP Authentication: Basic and Digest Access Authentication”, J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, L. Stewart, June 1999. URL: http://www.rfc-editor.org/rfc/rfc2617.txt

[STD0006] “User Datagram Protocol”, J. Postel, August 1980. URL: http://www.rfc-editor.org/rfc/std/std6.txt

[STD0007] “Transmission Control Protocol”, J. Postel, September 1981. URL: http://www.rfc-editor.org/rfc/std/std7.txt

[STD0013] “Domain Name System”, P. Mockapetris, November 1987. URL: http://www.rfc-editor.org/rfc/std/std13.txt

[TransportE2ESec] “WAP Transport Layer End-to-End Security Specification”, WAP Forum.

[UAProf] “User Agent Profile Specification”, WAP Forum.

[WAE] "Wireless Application Environment Specification", Version 2, WAP-236-WAESpec, WAP Forum.

[WAPCert] “WAP Certificate and CRL Profiles”, WAP Forum.

[WDP] "Wireless Datagram Protocol Specification", WAP Forum.

[WIM] “WAP Identity Module Specification”, WAP Forum.

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

[WML] "Wireless Markup Language", WAP Forum.

[WMLScriptCrypto] “WMLScript Crypto Library”, WAP Forum.

[WPKI] "WAP Public Key Infrastructure Definition", WAP Forum.

[WSP] "Wireless Session Protocol", WAP Forum.

[WTA] “Wireless Telephony Application Specification", WAP Forum.

[WTAI] "Wireless Telephony Application Interface", WAP Forum.

[WP-TCP] “Wireless Profiled TCP Specification”, WAP Forum.

[WTLS] "Wireless Transport Layer Security Protocol", WAP Forum.

[WTP] "Wireless Transaction Protocol Specification", WAP Forum.

[XHTML] “XHTML 1.1 – Module Based XHTML”, World Wide Web Consortium.
URL: http://www.w3.org/TR/xhtml11/

[XML] “Extensible Markup Language (XML) 1.0”, World Wide Web Consortium.
URL: http://www.w3.org/TR/REC-xml/

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

4. Definitions and Abbreviations

4.1 Definitions

The following are terms and conventions used throughout this specification.

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described by [RFC2119].

Author – an author is a person or program that writes or generates WML, WMLScript or other content.

Client – a device (or application) that initiates a request for a connection with a server.

Content – subject matter (data) stored or generated at an origin server. Content is typically displayed or interpreted by a user agent in response to a user request.

Content Encoding – when used as a verb, content encoding indicates the act of converting content from one format to another. Typically the resulting format requires less physical space than the original, is easier to process or store and/or is encrypted. When used as a noun, content encoding specifies a particular format or encoding standard or process.

Content Format – actual representation of content.

Device – a network entity that is capable of sending and receiving packets of information and has a unique device address. A device can act as both a client or a server within a given context or across multiple contexts. For example, a device can service a number of clients (as a server) while being a client to another server.

JavaScript – a *de facto* standard language that can be used to add dynamic behaviour to HTML documents. JavaScript is one of the originating technologies of ECMAScript.

Man-Machine Interface – a synonym for user interface.

Origin Server – the server on which a given resource resides or is to be created. Often referred to as a web server or an HTTP server.

Resource – a network data object or service that can be identified by a URI or URL. Resources may be available in multiple representations (e.g., multiple languages, data formats, size and resolutions) or vary in other ways.

Server – a device (or application) that passively waits for connection requests from one or more clients. A server may accept or reject a connection request from a client.

Terminal – a device providing the user with user agent capabilities, including the ability to request and receive information. Also called a mobile terminal or mobile station.

User – a user is a person who interacts with a user agent to view, hear, or otherwise use a resource.

User Agent – a user agent is any software or device that interprets WML, WMLScript, WTAI or other resources. This may include textual browsers, voice browsers, search engines, etc.

WMLScript – a scripting language used to program the mobile device. WMLScript is based on ECMAScript and loosely based on the JavaScript™ scripting languages.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

4.2 Abbreviations

For the purposes of this specification, the following abbreviations apply.

CGI Common Gateway Interface

CPU Central Processing Unit

DNS Domain Name System

EFI External Functionality Interface

HTML HyperText Markup Language [HTML4]

HTTP HyperText Transfer Protocol [RFC2616]

IP Internet Protocol

MMI Man-Machine Interface

MMS Multimedia Message Service

OTA Over The Air

PDA Personal Digital Assistant

PICS Protocol Implementation Conformance Statement

PKI Public Key Infrastructure

RAM Random Access Memory

RFC Request For Comments

ROM Read Only Memory

SCR Static Conformance Requirement

SSL Secure Sockets Layer

STD Internet Standard

TCP Transmission Control Protocol [STD0007]

TLS Transport Layer Security

UDP User Datagram Protocol [STD0006]

URI Uniform Resource Identifier [RFC2396]

URL Uniform Resource Locator [RFC2396]

W3C World Wide Web Consortium

WAE Wireless Application Environment [WAE]

WAP Wireless Application Protocol

WDP Wireless Datagram Protocol [WDP]

WIM Wireless Identity Module [WIM]

WML Wireless Markup Language [WML]

WPKI Wireless Public Key Infrastructure [WPKI]

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

WSP Wireless Session Protocol [WSP]

WTA Wireless Telephony Application [WTA]

WTAI Wireless Telephony Application Interface [WTAI]

WTLS Wireless Transport Layer Security [WTLS]

WTP Wireless Transaction Protocol [WTP]

WWW World-Wide Web

XHTML Extensible Hypertext Markup Language [XHTML]

XML Extensible Markup Language [XML]

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

5. Background

5.1 Motivation

WAP is positioned at the convergence of three rapidly evolving network technologies, wireless data, telephony, and the Internet.

Both the wireless data market and the Internet are growing very quickly and are continuing to reach new customers. The explosive growth of the Internet has fuelled the creation of new and exciting information services.

Most of the original technology developed for the Internet has been designed for desktop and larger computers and medium to high bandwidth, generally reliable data networks. Mass-market, hand-held wireless devices present a more constrained computing environment compared to desktop computers. Because of fundamental limitations of power and form-factor, mass-market handheld devices tend to have:

- Less powerful CPUs,
- Less memory (ROM and RAM),
- Restricted power consumption,
- Smaller displays, and
- Different input devices (e.g., a phone keypad).

Similarly, wireless data networks present a more constrained communication environment compared to wired networks. Because of fundamental limitations of power, available spectrum, and mobility, wireless data networks tend to have:

- Less bandwidth,
- More latency,
- Less connection stability, and
- Less predictable availability.

Mobile networks are growing in complexity and the cost of all aspects of providing value-added services is increasing. In order to meet the requirements of mobile network operators, solutions must be:

- Interoperable – terminals from different manufacturers communicate with services in the mobile network;
- Scaleable – mobile network operators are able to scale services to customer needs;
- Efficient – provides quality of service suited to the behaviour and characteristics of the mobile network;
- Reliable – provides a consistent and predictable platform for deploying services; and
- Secure – enables services to be extended over potentially unprotected mobile networks while still preserving the integrity of user data; protects the devices and services from security problems such as loss of confidentiality.

Many of the current mobile networks include advanced services that can be offered to end-users. Mobile network operators strive to provide advanced services in a useable and attractive way in order to promote increased usage of the mobile network services and to decrease the turnover rate of subscribers. Standard features, like call control, can be enhanced by using WAP technology to provide customised user interfaces. For example, services such as call forwarding may provide a user interface that prompts the user to make a choice between accepting a call, forwarding to another person, forwarding it to voice mail, etc.

The nature of wireless devices is that they are inherently mobile. This mobility introduces new opportunities for services that are sensitive to mobility and can provide location-dependent information. The WAP specifications and architecture capitalise on this unique aspect of wireless devices by including mobility as part of the application model.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

The WAP specifications address mobile network characteristics and operator needs by adapting existing network technology to the special requirements of mass-market, hand-held wireless data devices and by introducing new technology where appropriate.

The WAP specifications will accommodate a range of devices, from devices that provide very basic functionality to devices that continue to expand their capabilities. This motivates an architecture where functionality may be moved to different locations within the network as appropriate, i.e. either to devices or to network servers as necessary.

5.2 Architectural Goals

The goals of the WAP Forum architecture are as follows. This summary is informative and non-exhaustive; the order of the items does not represent any priority or importance.

- Provide a web-centric application model for wireless data services that utilises the telephony, mobility, and other unique functions of wireless devices and networks and allows maximum flexibility and ability for vendors to enhance the user experience.
- Enable the personalisation and customisation of the device, the content delivered to it, and the presentation of the content.
- Provide support for secure and private applications and communication in a manner that is consistent and interoperable with Internet security models.
- Enable wireless devices and networks that are currently or in the near future being deployed, including a wide variety of bearers from narrow-band to wide-band.
- Provide secure access to local handset functionality.
- Facilitate network-operator and third party service provisioning.
- Define a layered, scaleable and extensible architecture.
- Leverage existing standards where possible, especially existing and evolving Internet standards.

6. Architecture Overview

6.1 The World-Wide Web Model

The Internet World-Wide Web (WWW) architecture provides a very flexible and powerful programming model (Figure 1). 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, i.e., it sends requests for named data objects to a network server and the network server responds with the data encoded using the standard formats.

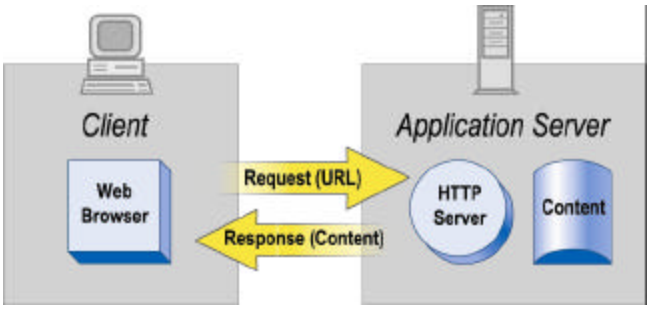

Figure 1. World-Wide-Web Programming Model

The WWW standards specify many of the mechanisms necessary to build a general-purpose application environment, including:

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

- Standard naming model – All servers and content on the WWW are named with an Internet-standard *Uniform Resource Locator* (URL) [RFC2396].
- 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 [RFC2045, RFC2048].
- Standard content formats – All web browsers support a set of standard content formats. These include the HyperText Markup Language (HTML) [HTML4], scripting languages [ECMAScript, JavaScript], and a large number of other formats.
- 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 HyperText Transport Protocol (HTTP) [RFC2616], operating on top of the TCP/IP protocol suite [STD0007].

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.

6.2 The WAP Model

The WAP programming model (Figure 2) is the WWW programming model with a few enhancements. Adopting the WWW programming model provides several benefits to the application developer community, including a familiar programming model, a proven architecture, and the ability to leverage existing tools (e.g., Web servers, XML tools, etc.). Optimisations and extensions have been made in order to match the characteristics of the wireless environment. Wherever possible, existing standards have been adopted or have been used as the starting point for the WAP technology.

The most significant enhancements WAP has added to the programming model are:

- Push
- Telephony Support (WTA)

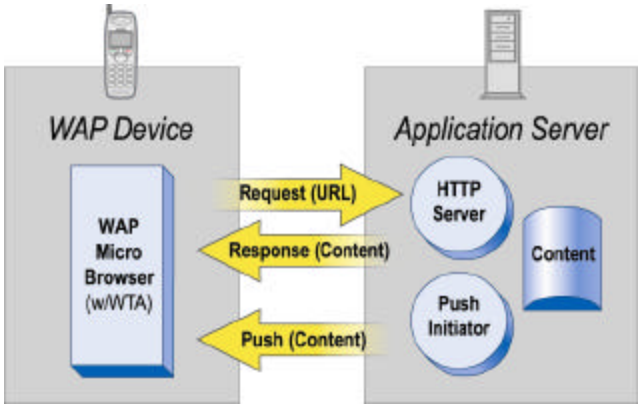

Figure 2. WAP Programming Model

The classical request-response mechanism is commonly referred to as *pull* to contrast it with the *push* mechanism.

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 co-ordinates 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:

- Standard naming model – WWW-standard URLs are used to identify WAP content on origin servers. WWW-standard URIs are used to identify local resources in a device, e.g. call control functions.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

- Content typing – All WAP content is given a specific type consistent with WWW typing. This allows WAP user agents to correctly process the content based on its type.
- Standard content formats – WAP content formats are based on WWW technology and include display markup, calendar information, electronic business card objects, images and scripting language.
- Standard communication protocols – WAP communication protocols enable the communication of browser requests from the mobile terminal to the network web server.

The WAP content types and protocols have been optimised for mass market, hand-held wireless devices.

6.3 Feature/Performance-Enhancing Proxies

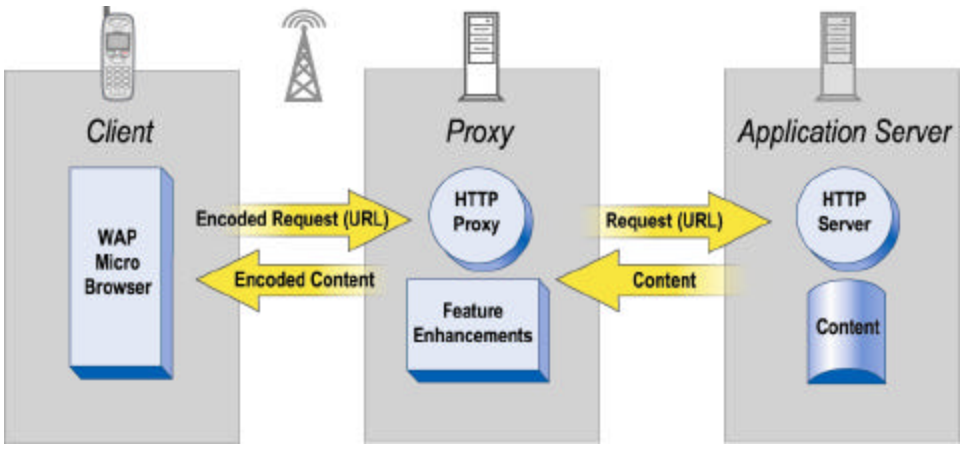

Figure 3. Feature/Performance-Enhancing Proxy

WAP utilises proxy technology to optimise and enhance the connection between the wireless domain and the WWW. The WAP proxy may provide a variety of functions, including:

- Protocol Gateway – The protocol gateway translates requests from a wireless protocol stack (e.g., the WAP 1.x stack—WSP, WTP, WTLS, and WDP) to the WWW protocols (HTTP and TCP/IP). The gateway also performs DNS lookups of the servers named by the client in the request URLs.
- Content Encoders and Decoders – The content encoders can be used to translate WAP content into a compact format that allows for better utilisation of the underlying link due to its reduced size.
- User Agent Profile Management – User agent profiles describing client capabilities and personal preferences [UAProf] are composed and presented to the applications.
- Caching Proxy – A caching proxy can improve perceived performance and network utilisation by maintaining a cache of frequently accessed resources.

This infrastructure ensures that mobile terminal users can access a wide variety of Internet content and applications, and that application authors are able to build content services and applications that run on a large base of mobile terminals. The WAP proxy allows content and applications to be hosted on standard WWW servers and to be developed using proven WWW technologies such as CGI scripting.

While the nominal use of WAP will include a web server, WAP proxy and WAP client, the WAP architecture can quite easily support other configurations.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

6.4 Supporting Servers

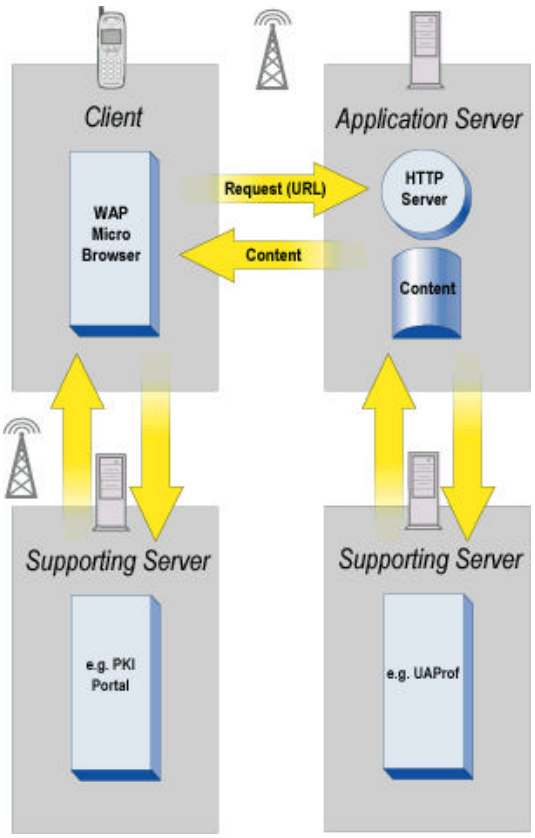

Figure 4. Supporting Services

The WAP Architecture also includes supporting servers, which provide services to devices, proxies, and applications as needed. These services are often specific in function, but are of general use to a wide variety of applications.

The supporting servers defined by the WAP Forum include, but are not limited to:

- PKI Portal—The PKI Portal (shown in Figure 4) [WPKI] allows devices to initiate the creation of new public key certificates.
- UAProf Server—The UAProf Server [UAProf] allows applications to retrieve the client capabilities and personal profiles of user agents and individual users.
- Provisioning Server—The Provisioning Server [ProvArch] is trusted by the WAP device to provide its provisioning information.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

6.5 WAP Network Elements

A typical WAP network is shown in Figure 5.

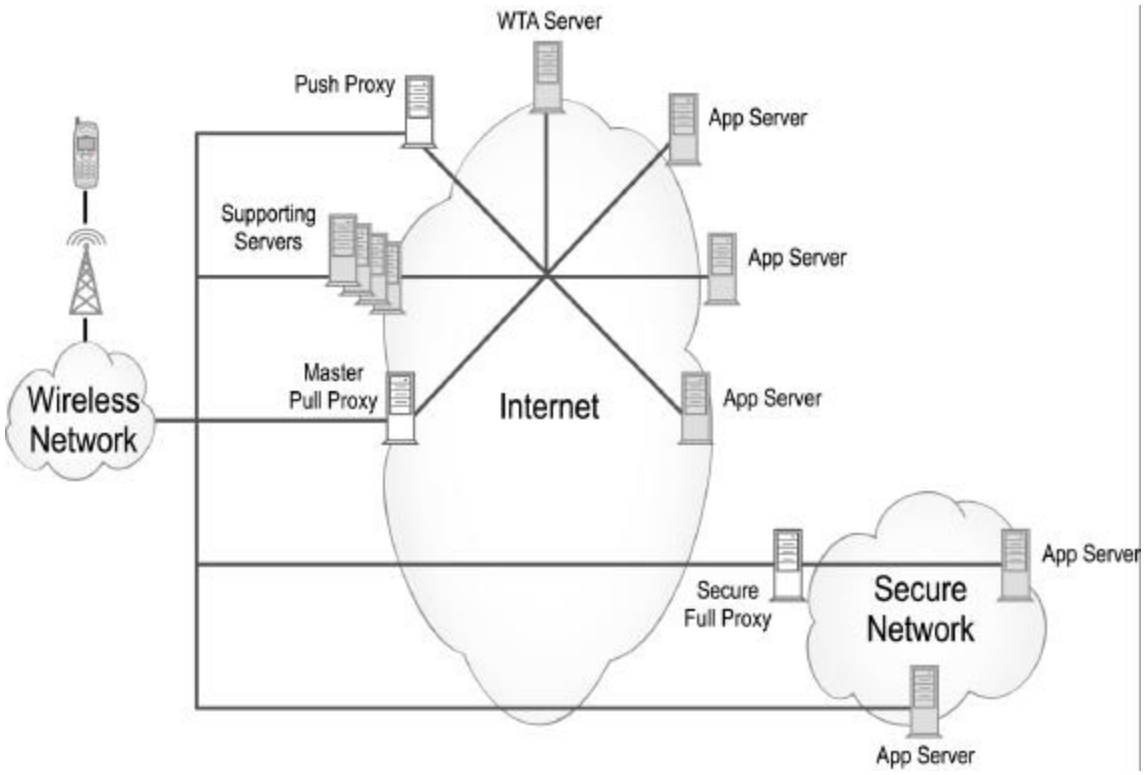

Figure 5. Example WAP Network

WAP clients communicate with application servers through a number of different proxies or directly. WAP clients support the *proxy selection* mechanism that allows them to utilise the most appropriate proxy for a given service or to connect directly to that service as necessary. Proxies can be used to augment a request. They translate between WAP and WWW protocols (HTTP, TCP), thereby allowing the WAP client to submit requests to the origin server.

Proxies may be located in a number of places, including wireless carriers or independent service providers in order to provide feature enhancements coupled to the wireless network (e.g., telephony, location and provisioning) or to optimise the communication between device and application server (e.g., protocol translation and cookie caching). Proxies may be located in a secure network to provide a secure channel between wireless device and the secure network.

In some instances, the device might make direct connections to application servers, for example to provide a secure connection directly between the device and application server.

The supporting servers provide support functions required by or generally useful to devices, proxies, and application servers. These functions include Provisioning, PKI, user agent profiles, etc.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

6.6 Device Architecture

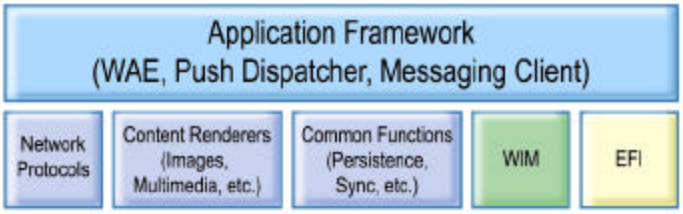

Figure 6. WAP Client Architecture

The architecture for WAP devices is shown in Figure 6. The Application Framework provides the device execution environment for WAP applications. WAP applications are comprised of markup, script, style sheets and multimedia content, all of which are rendered on the device. The WAP Application Environment (WAE) processing model defines the structure in which these various forms of executable and non-executable content interact.

The network protocols on the WAP client are shared between client and server. They are described in further detail below. Content renderers interpret specific forms of content and present them to the end user for perusal or interaction. Common functions are defined to be utilised by the application framework, including persistence and data synchronisation.

The Wireless Identity Module (WIM), as specified in [WIM], contains the identity of the device and the cryptographic means to mutually authenticate WAP devices and servers.

The architecture also provides a mechanism to access external functions that are embedded or attached to the devices via the External Functionality Interface (EFI).

6.7 Security Model

WAP enables a flexible security infrastructure that focuses on providing connection security between a WAP client and server.

WAP can provide end-to-end security between protocol endpoints. If a browser and origin server desire end-to-end security, they can communicate directly using the security protocols. Moreover, the WAP specifications include support for application-level security, such as signed text.

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

7. Components of the WAP Architecture

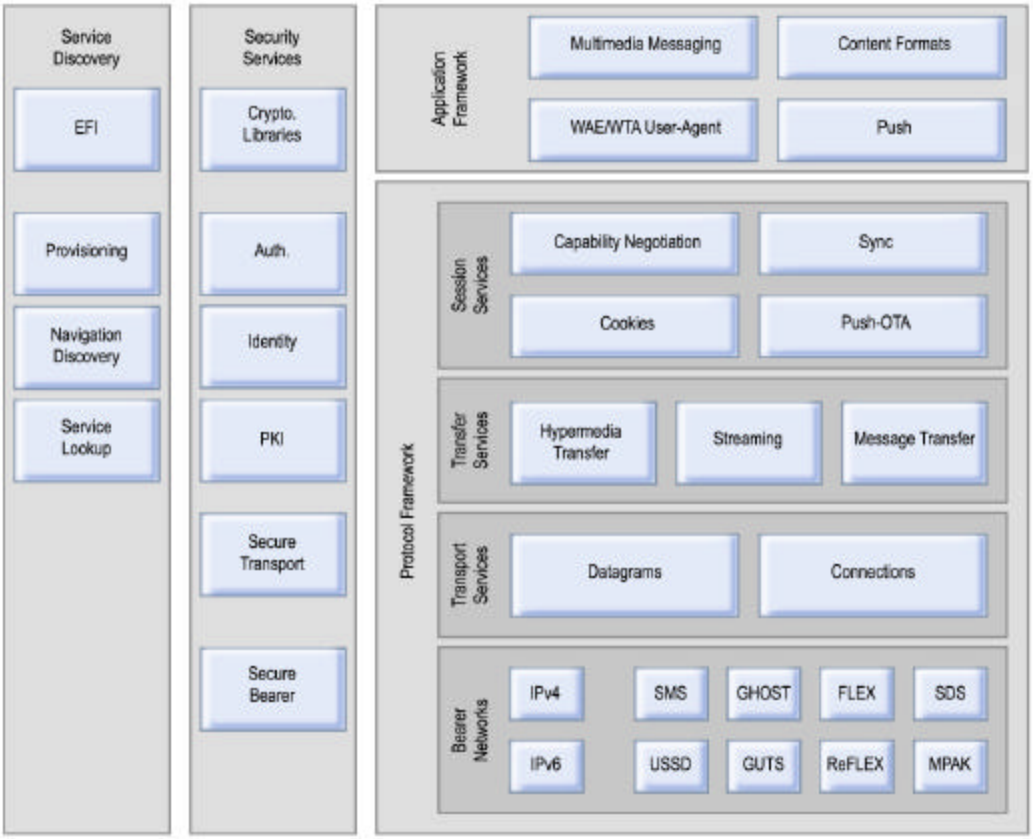

Figure 7. WAP Stack Architecture

The WAP architecture provides a scaleable and extensible application development environment for mobile communication devices. This is achieved through a layered design of the protocol stack (Figure 7). Each layer provides a set of functions and/or services to other services and applications through a set of well-defined interfaces. Each of the layers of the architecture is accessible by the layers above, as well as by other services and applications.

The WAP architecture separates service interfaces from the protocols that provide those services to allow for evolution of the specifications and selection of the most appropriate protocol for a given context. Many of the services in the stack may be provided by more than one protocol. For example, either HTTP [RFC2616] or WSP [WSP] may provide the Hypermedia Transfer service.

7.1 Bearer Networks

Protocols have either been designed or selected to operate over a variety of different bearer services, including short message, circuit-switched data, and packet data. The bearers offer differing levels of quality of service with respect to throughput, error rate, and delays. The protocols are designed to compensate for or tolerate these varying levels of service.

Since the Transport Services layer provides the interface between the bearer service and the rest of the WAP stack, the transport specifications (e.g., [WDP]) may list the bearers that are supported and the techniques used to allow the protocols to run over each bearer. The list of supported bearers will change over time with new bearers being added as the wireless market evolves.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

7.2 Transport Services

The Transport Services layer offers a set of consistent services to the upper layer protocols and maps those services to the available bearer services. The Transport Services transport unstructured data across the underlying bearer networks. These transport services create a common abstraction that is consistent across all the bearers.

The Transport Services include, but are not limited to:

- Datagrams – The datagram service provides data transport in which self-contained, independent entities of data carry sufficient information to be routed from the source to the destination computer without reliance on earlier exchanges between this source and destination computer and the transporting network. UDP (User Datagram Protocol) [STD0006] and WDP (Wireless Datagram Protocol) [WDP] are two protocols used to provide the datagram transport service in the WAP architecture.
- Connections – The connection service provides data transport service in which communication proceeds in three well-defined phases: connection establishment, two-way reliable data transfer and connection release. TCP (Transmission Control Protocol) [STD0007] is a protocol used to provide the connection transport service of IP[1] bearers for the WAP architecture. In order to cope with the wireless network characteristics, the TCP protocol can be profiled for its use, see [WP-TCP].

7.3 Transfer Services

The Transfer Services provide for the structured transfer of information between network elements.

The Transfer Services include, but are not limited to:

- Hypermedia Transfer – The hypermedia transfer services provides for the transfer of self-describing hypermedia resources. The combination of WSP (Wireless Session Protocol) [WSP] and WTP (Wireless Transaction Protocol) [WTP] provide the hypermedia transfer service over secure and non-secure datagram transports. The HTTP (Hypertext Transfer Protocol) [RFC2616] provides the hypermedia transfer service over secure and non-secure connection-oriented transports.
- Streaming – The streaming services provide a means for transferring isochronous data such as audio and video.
- Message Transfer – The message transfer services provide the means to transfer asynchronous multimedia messages such as email or instant messages. MMS Encapsulation [MMSEncapsulation] is a protocol used to transfer messages between WAP devices and MMS servers.

7.4 Session Services

The session services provide for the establishment of shared state between network elements that span multiple network requests or data transfers. For example, the Push session establishes that the WAP Device is ready and able to receive pushes from the Push Proxy.

The Session Services include, but are not limited to:

- Capability Negotiation – The WAP architecture includes specifications for describing, transmitting, and managing capabilities and preference information about the client, user, and network elements. See [UAProf] for more information. This allows for customisation of information and content returned by the origin server or pushed by the application.
- Push-OTA – The Push-OTA (Over The Air) session service provides for network-initiated transactions to be delivered to wireless devices that are intermittently able to receive data (e.g., modal devices and devices with dynamically assigned addresses). The Push-OTA service operates over the connection-oriented transport service and datagram transport [PushOTA].
- Sync – The Sync service provides for the synchronisation of replicated data.

---

[1] Utilisation of TCP connections over IP may require additional components of the TCP/IP protocol suite. One example for such a component is ICMP.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

- Cookies – The Cookies service allows applications to establish state on the client or proxy that survives multiple hypermedia transfer transactions. See [HTTPState] for more information.

7.5 Application Framework

The Application Framework provides a general-purpose application environment based on a combination of World Wide Web (WWW), Internet and Mobile Telephony technologies. The primary objective of the Application Framework is to establish an interoperable environment that will allow operators and service providers to build applications and services that can reach a wide variety of different wireless platforms in an efficient and useful manner.

The Application Framework includes, but is not limited to:

- WAE/WTA User-Agent – WAE is a micro-browser environment containing or allowing for markup (including WML and XHTML), scripting, style-sheet languages, and telephony services and programming interfaces, all optimised for use in hand-held mobile terminals. See [WAE] for more information.
- Push – The Push service provides a general mechanism for the network to initiate the transmission of data to applications resident on WAP devices. See [PushArchOverview] for more information.
- Multimedia Messaging – The Multimedia Message Service (MMS) provides for the transfer and processing of multimedia messages such as email and instant messages to WAP devices.
- Content Formats – The application framework includes support for a set of well-defined data formats, such as color images, audio, video, animation, phone book records, and calendar information.

7.6 Security Services

Security forms a fundamental part of the WAP Architecture, and its services can be found in many of its layers. In general the following security facilities offered are:

- Privacy – facilities to ensure that communication is private and cannot be understood by any intermediate parties that may have intercepted it.
- Authentication – facilities to establish the authenticity of parties to the communication.
- Integrity – facilities to ensure that communication is unchanged and uncorrupted.
- Non-Repudiation – facilities to ensure parties to a communication cannot deny the communication took place.

The Security Services span all the various layers of the WAP Architecture. Some specific examples of the security services include:

- Cryptographic Libraries – This application framework level library provides services for signing of data for integrity and non-repudiation purposes. See [WMLScriptCrypto] for more information.
- Authentication – The Security Services provide various mechanisms for client and server authentication. At the Session Services layer HTTP Client Authentication [RFC2617] may be used to authenticate clients to proxies and application servers. At the Transport Services layer, WTLS and TLS handshakes may be used to authenticate clients and servers.
- Identity – WIM provides the functions that store and process information needed for user identification and authentication [WIM]
- PKI – The set of security services that enable the use and management of public-key cryptography and certificates [WPKI], [WAPCert].
- Secure Transport – The Transport Services layer protocols are defined for secure transport over datagrams and connections. WTLS is defined for secure transport over datagrams and TLS is defined for secure transport over connections (i.e. TCP). See [WTLS] and [WAPTLS] for more information.
- Secure Bearer – Some bearer networks provide bearer level security. For example, IP networks (especially in the context of IPv6) provide bearer-level security with IPSec [RFC2401].

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

7.7 Service Discovery

Service discovery forms a fundamental part of the WAP Architecture and its services can be found at many layers. Some specific examples of Service Discovery services include:

- EFI – The External Functionality Interface (EFI) allows applications to discover what external functions/services are available on the device.
- Provisioning – The Provisioning service allows a device to be provisioned with the parameters necessary to access network services. See [ProvArch] for more information.
- Navigation Discovery – The Navigation Discovery service allows a device to discover new network services (e.g. secure pull proxies) during the course of navigation such as when downloading resources from a hypermedia server. The WAP Transport-Level End-to-End Security specification [TransportE2ESec] defines one navigation discovery protocol.
- Service Lookup – The Service Lookup service provides for the discovery of a service's parameters through a directory lookup by name. One example of this is the Domain Name System (DNS) [STD0013].

7.8 Other Services and Applications

The WAP layered architecture enables other services and applications to utilise the features of the WAP stack through a set of well-defined interfaces. External applications may access the various services directly. The WAP layered architecture builds upon an extensible set of protocols. This allows the WAP stack to be used for applications and services not currently specified by WAP, but deemed to be valuable for the wireless market. Such applications and services may benefit from adding protocols or particular protocol capabilities. For example, applications, such as electronic mail, calendar, phone book, notepad, and electronic commerce, or services, such as white and yellow pages, may be developed to use the WAP protocols.

8. Sample Configurations of WAP Technology

Because several of the services in the WAP stack can be provided using different protocols based on the circumstances, there are more than one possible stack configurations. The following figures depict several possible protocol stacks using WAP technology. These are for illustrative, informative purposes only and do not constitute a statement of conformance or interoperability, nor is this set of examples exhaustive.

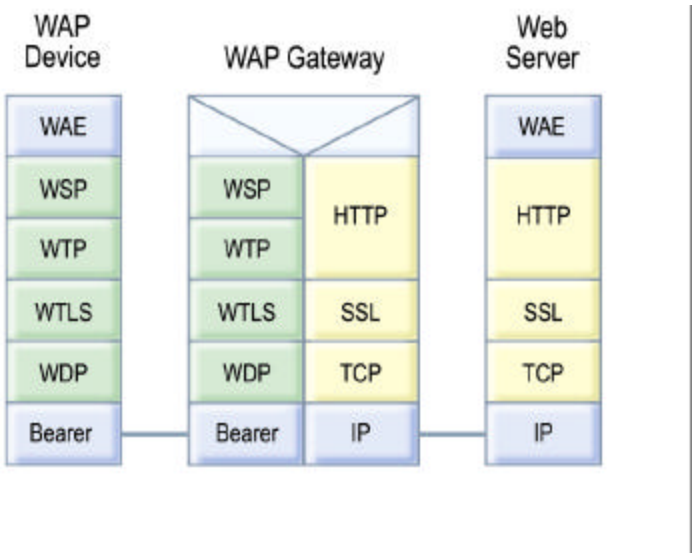

Figure 8. Example WAP 1.x Gateway

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

Figure 8 depicts the protocol stacks for the original WAP architecture. The WAP Gateway converts the hypermedia transfer service between the datagram-based protocols (WSP, WTP, WTLS, WDP) and connection-oriented protocols commonly used on the Internet (HTTP, SSL, TCP).

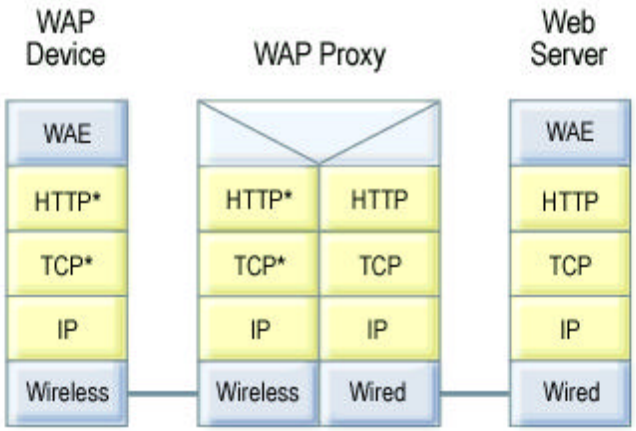

Figure 9. Example WAP HTTP Proxy with Profiled TCP and HTTP

Figure 9 depicts a WAP HTTP proxy. The proxy configuration is widely used in the Internet for ordinary web access, multimedia data, e.g. music, video clip downloading and so on. This configuration locates the WAP Proxy between wireline and wireless networks to enhance performance by using the wireless profile of TCP (as shown with TCP*). In addition to TCP optimisations, the wireless profile of HTTP (as illustrated by HTTP*) allows for further performance enhancements. Both profiles comprise well-defined IETF options that provide for efficient operation over wireless networks as within the scope of WAP. The wireless profiled versions are interoperable with TCP and HTTP.

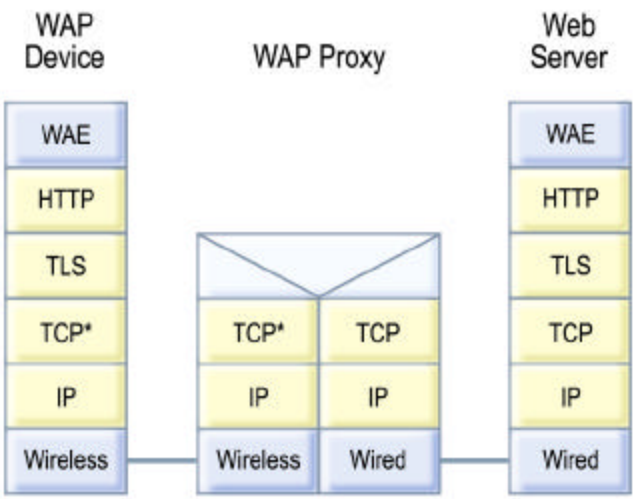

Figure 10. Example WAP Proxy Support for TLS Tunneling

Ó Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

Figure 10 depicts a WAP HTTP proxy that has established a connection-oriented tunnel to the web server (e.g., in response to a CONNECT command). This configuration is used to allow TLS to provide end-to-end security between mobile terminal and origin server. E-commerce is a compelling use case for end-to-end security.

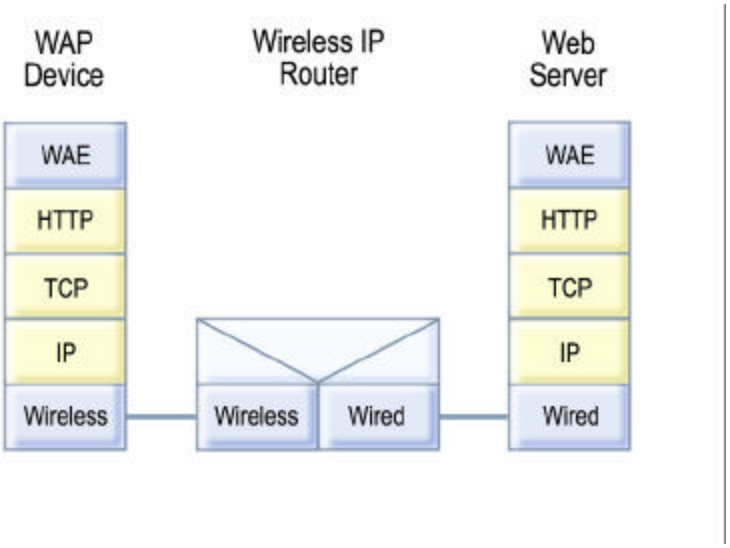

Figure 11. Example Direct Access

Figure 11 depicts a WAP device directly accessing a Web Server via the Internet. The wireless IP router is a standard part of an IP network that is used to transfer IP packets from one link layer (e.g., the wireless link) to another (e.g., the wired link). This configuration can apply to the case where bearer-level security (such as IPSec) is utilised.

In the Direct Access scenario, wireless optimisations as defined by the Wireless Profiles for TCP and HTTP may not be available.

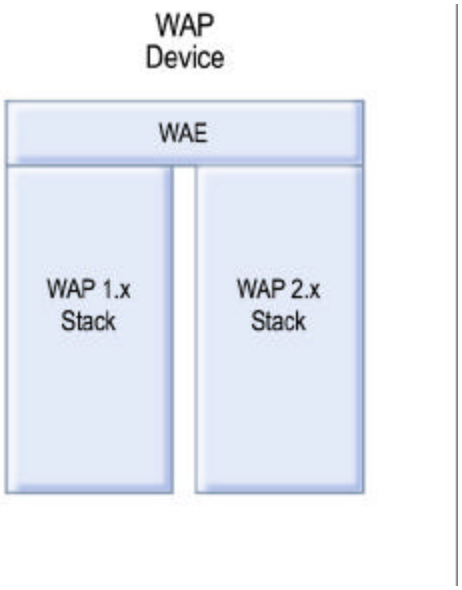

Figure 12. Dual Stack Support

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved

While the previous configurations show single protocol stacks for each of WAP configuration, Figure 12 depicts a device that supports both the 1.x and 2.x protocol stacks. This configuration is useful in cases where a device needs to interoperate with both old a new WAP servers.

9. Conformance and Interoperability

The WAP Forum views vendor interoperability as an important element to the success of WAP products. In order to provide as high a probability as is technically possible that two WAP products developed independently by two different vendors will successfully interoperate, a rigorous definition of conformance, compliance, and testing has been developed.

Conformance answers the question, "Does an implementation meet the standard as written?" The WAP Forum charters a neutral third party to build a comprehensive test suite from its specifications. Usually, implementations are tested against known references.

Interoperability answers the question, "Will this implementation work with other products developed to the same standard?" Interoperability testing uses a test suite designed to test implementation to implementation compatibility, and implementations are tested against each other. Interoperability testing is not focused on compliance—two products with the same non-compliant implementation will be interoperable.

Figure 13. Interoperability and Compliance

The WAP Forum Certification Program is focused on conformance, but offers some interoperability testing as well. The Certification Program covers the entire value chain as shown in Figure 13.

To improve interoperability at the authoring level, the WAP Forum provides authoring guidelines to improve the accessibility of WAP content. To certify WAP clients and servers, the WAP Forum conducts interoperability testing of an implementation against multiple reference implementations using a predefined suite of test cases.

The WAP Forum has defined a number of Class Conformance Profiles, e.g. Class A, Class B, and Class C. An implementation may be certified in one or more class. The class conformance requirements are specified in [ClassConform].

Each WAP Specification includes static conformance requirements (SCRs) for that specification. These define which features are mandatory and optional and are the basis for the conformance test suite.

© Copyright Wireless Application Protocol Forum, Ltd, 2000-2001
All rights reserved