The Future of WAP: v1.2 and Beyond

Andy Wigley, Secure Trading

Currently, the WAP browsers and gateways available in the marketplace support WAP v1.1 specifications. However, WAP v1.2 was approved by the WAP Forum in November 1999, and a minor revision (v1.2.1) has now been released too. Browsers and gateways supporting the functionality it defines should reach the market before the end of 2000.

WAP v1.2 has the same functionality as WAP v1.1, and continues the incorporation of detailed feedback from implementers. The main thrust of v1.2 has been to remove ambiguities and to make minor editorial corrections to the v1.1 specifications. At present, developers all too often find themselves in the unsatisfactory position of coding to the 'lowest common denominator', in order to address the presentational differences between, for example, Nokia and Phone.com browsers.

New Features in v1.2

WAP v1.2 introduces the WAP Push Architecture, enhances the Wireless Telephony Applications (WTA) specifications, includes support for additional bearer network technologies, and introduces the Wireless Identity Module (WIM), which paves the way for wireless security with a strength exceeding that commonly in use for wired e-commerce.

These enhancements should accelerate WAP acceptance among consumers. Push allows a new breed of WAP application to be developed, where alerts and trigger messages can be delivered to consumers, thus delivering information without the consumer having to specifically go and retrieve it. The security enhancements should go a long way to engendering trust in consumers for m-commerce and will be welcomed by retailers, as it makes financial fraud very much harder to achieve. Developers will be in the happy position of being able to develop applications where the end user has a communications device capable of cryptographic operations, and which can receive data in an unsolicited fashion — a situation not often encountered elsewhere in web development.

One big pointer to WAP's (and the Web's) future is the specification of the User Agent Profile (UAProf), which we will see to be a crucial ingredient for W3C convergence.

WML Changes

The changes to WML that will be of interest to developers are, thankfully, fairly minor. In general, they address inconsistencies that have become apparent in early WAP implementations and aim to give the user a more consistent, user-friendly interface.

accesskey attribute

This attribute assigns an access key to an element. Its purpose is to allow the user to activate a particular element by using a single key, enabling developers to build 'shortcuts' into their applications.

The keys available will vary depending on the type of mobile device being used. (Phones, for example, will usually have 0-9, * and # keys.) For this reason, the user agent is not *required* to support accesskey.

<pre> element

The <pre> element tells visual user agents that the enclosed text is preformatted. When handling preformatted text, user agents should make a "best effort" to:

- Leave white space intact
- Render text with a fixed-pitch font
- Disable automatic word wrap

This feature allows the developer to override any presentational differences between different browsers, giving users a more consistent interface regardless of the end device used.

<table> default alignment attribute

This allows the WML author a formal means of specifying the default alignment for a table, rather than — as at present — leaving it to chance. Default alignment is applied to columns that are missing alignment designators, or have unrecognized designators. This gives the developer more direct control over the rendering of tabulated data.

<go> enctype attribute

This attribute allows the author to specify how the user agent should submit data on a POST operation. In WAP v1.2, only `application/x-www-form-urlencoded` (the default) or `multipart/form-data` can be specified.

<input> format semantics changed

For example, `<input name='X' format= "4N">` now means 0–4 numeric characters. In v1.1, this often meant precisely 4 characters, although there were differences in interpretation between the different browsers.

WTAI Enhancements

Most of the work in enhancing the Wireless Telephony Application Interface has been in removing ambiguities and correcting problems in the version 1.1 specifications. Currently, support in WML browsers for WTAI is not wide. However, if you're building applications for the more feature-rich browsers, the following are some of the more significant changes:

- Mandatory display of the number in public `makeCall(`*`number`*`)` function
- New functions for:
  - Adding a new phonebook entry: `addPBEntry(`*`number, name`*`)`
  - Voice call control: `callStatus(`*`id, field`*`), listCall(`*`id`*`)`
  - Changing a phonebook entry: `change(`*`id, field, value`*`)`
- New events for:
  - Voice call control: Outgoing Call indication (`cc/oc`), Connected Call indication (`cc/co`), DTMF sent (`cc/dtmf`)
  - Incoming network text (`nt/it`), and network text sent (`nt/st`)
  - Network status indication (`ms/ns`)

New Push Specifications

Push is a major new piece of functionality that is a significant addition to the WAP developer's toolkit. It's a good example of how the WAP Forum is endeavoring to create web connectivity tailored to the different needs of wireless subscribers, rather than just "bringing the Web to your pocket".

The WAP push framework introduces a means within the WAP effort to transmit information to a device without a previous user action. In the normal client-server model, a client requests a service or information from a server, which then responds by transmitting information to the client. This is known as "pull" technology — the client "pulls" information from the server. The Web is a typical example of pull technology: a user enters a URL (the request) that's sent to a server, and the server answers by sending a web page (the response) to the user.

In contrast to this is "push" technology, which is also based on the client-server model, but involves no explicit request from the client before the server transmits its content.

As with standard WAP architecture, a proxy lying between the client(s) and the content server performs a crucial role. It may be that the standard WAP "pull" proxy (that is, the WAP gateway) may be the PPG (push proxy gateway) as well, but this is not a requirement.

The PPG acts as a store-and-forward agent for push submissions, and may return status information to the push-initiating content server when the operation has been completed. Push submissions may be directed to a single client, or some PPGs may implement an address aliasing scheme whereby a single push submission addressed to a special destination may, in fact, reach a group of clients.

What will Push be used for?

Typical applications for push technology will be those where a user has requested notifications, say, for new e-mail, stock price movements, goals scored at football games, and so on.

At its simplest, a push message will consist of a short message and a URL that the user will select for further information. However, one of the more interesting aspects of the push architecture is that it does not dictate that the content must be WML or WMLScript. The header in a push message includes an **application identifier** that indicates the particular user agent (or application) the content is targeted at. This raises the interesting prospect of WAP being used as the transport mechanism for content intended for any other applications installed on the mobile device.

Push Security

Needless to say, feeding content to a device without the specific request of the client requires careful consideration of security. The WAP specifications provide a lot of ideas about how this is to be achieved, but leaves it to the implementers to decide exactly what approach should be followed.

As a general principle, of course, the content server initiating the push *must* be authenticated in one form or another. This could be through TLS/SSL certificates between the PPG and push initiator, or by using object-level certificates to encrypt and digitally sign submissions, so that end-to-end security is used and the PPG is not involved in the authentication. Digest-based HTTP authentication could also be used.

In certain circumstances, clients can delegate trust to the PPG. The client can maintain a list of trusted PPGs, and as long as the push content is being submitted by one of those, the client can accept that the PPG has in turn authenticated the push initiator.

User Agent Profile

The User Agent Profile (UAProf) is included in the v1.2 specifications, but its true significance will only become apparent as the Web moves towards **convergence** (of which more later). It is placing a very large stake in the ground about the future of WAP and the Web.

The UAProf specification extends WAP 1.1 to enable the end-to-end flow of a user agent profile (also referred to as *Capability and Preference Information* or *CPI*) between the WAP client, intermediate network points, and the origin server. It seeks to interoperate seamlessly with the emerging standards for Composite Capability/Preference Profile (CC/PP) distribution over the Internet. The specification defines a set of components and attributes that WAP-enabled devices may convey within the CPI. These may include, but are not limited to, the following:

- Hardware characteristics (screen size, color capabilities, image capabilities, manufacturer, etc.)
- Software characteristics (operating system vendor and version, support for MExE, list of audio and video encoders, etc.)
- Application/user preferences (browser manufacturer and version, markup languages and versions supported, scripting languages supported, etc.)
- WAP characteristics (WMLScript libraries, WAP version, WML deck size, etc.)
- Network characteristics (bearer characteristics such as latency and reliability, etc.)

With this capability, information about a client device's capabilities is available at the content server, so that application developers can tailor content accordingly. The W3C's vision is that, in time, the majority of web content will be defined once, with rendering of that content according to the end device's capabilities carried out as a separate step.

Instead of developing content aimed at a single class of end user devices, for example PCs, Wireless devices or TVs, content can be defined once and made available on many devices. This will lead to wider availability of content to consumers, regardless of the client device used.

Wireless Security Enhancements

For those of us interested in wireless m-commerce (and let's face it — who isn't?), the enhancements to wireless security are the most interesting part of WAP v1.2. They provide for additional, robust, application-level security, over and above that specified in version 1.1.

In v1.1, WTLS provides the capability to generate Master secrets, which are used as a source of entropy, to calculate MAC (Message Authentication Code) keys and message encryption keys that are used to secure a limited number of messages, depending on usage of WTLS. It also allows for server and client authentication.

In v1.2, the WIM allows secure storage and use of permanent private keys. It provides secure storage of the Master secrets defined for WAP 1.1, and the client keys used for client authentication. In addition, using the new WMLScript crypto library, it allows for digital signatures to be generated for use in end-to-end security at the application level.

Crucial to this is a tamper-proof module in the mobile device called the WIM (Wireless Identity Module). This has been defined with smart cards in mind, although the physical packaging of the WIM has been left to the discretion of manufacturers. Often it will be combined with the existing mobile phone SIM (which is, after all, a kind of smart card), giving rise to the rather unfortunately named SWIM.

For optimum security, some parts of the security functionality *need* to be performed by a tamper-resistant device, so that an attacker cannot retrieve sensitive data. This especially applies to data such as the permanent private keys used in the WTLS handshake with client authentication, and for making application-level electronic signatures (such as confirming an application-level transaction).

The WIM uses generic cryptographic features, so it could also be used for non-WAP applications, like SSL, TLS, S/MIME, etc.

The WAP v1.1 Security "Loophole"

There is strong pressure within the industry for the adoption of public key infrastructure (PKI) as the way of securing Internet (and particularly wireless) e-commerce.

Banking and commercial organizations are generally not satisfied with the security offered by WAP v1.1. Some of the reasons for this are due to the patchy levels of implementation that are characteristic of an emerging technology — some of the first WAP phones available on the market didn't support WTLS encryption at all. Of those that do, at the time of presenting only one (the Nokia 7110) does any WAP gateway authentication — that is, the ability to request a WAP gateway's certificate, authenticate it, and ask the consumer whether it should be accepted. All of the other phones that support WTLS use it only for encryption, not for the additional safeguard of authentication.

Furthermore, many of the WAP gateways deployed early on didn't support WTLS either, and some didn't even support TLS/SSL interaction with the content servers. However, these problems will surely disappear as the industry develops.

Even when the above problems are resolved, though, there still remains the issue of the role of the WAP gateway. This has the job of linking the WAP protocols on the wireless side to the web protocols of the wired world. For a secure data transfer, this requires that the WTLS-encrypted data sent from the WAP client must be momentarily decrypted to clear text, then immediately encrypted for onward transmission to the content servers over TLS/SSL.

Since this only happens for an instant, and WAP gateways tend to be operated by mobile phone network operators (which we would like to think are trusted organizations), it could be argued that to all intents and purposes, this *is* a secure connection. However, that can't be absolutely guaranteed, and consequently the gateway represents the most vulnerable link in an end-to-end transaction.

WAP v1.2 Application-level Security

With the WIM, private encryption keys can be stored at the device, while the corresponding public keys are made available to recipients. With this arrangement, it's possible to add a digital signature to messages, which provides additional end-to-end encryption (and hence integrity), and also provides *for non-repudiation*.

Applications such as e-commerce require the ability to provide persistent proof that someone has authorized a transaction. Although WTLS provides transient client authentication for the duration of a WTLS connection, it does not provide persistent authentication for transactions that may occur during that connection.

To support this requirement, the browser provides a WMLScript function, `Crypto.signText()`, that asks the user to sign a string of text. A call to the method displays the exact text to be signed and asks the user to confirm it. After the data has been signed and both the signature and the data have been sent across the network, the server can then extract the digital signature and validate it, and possibly store it for accountability purposes.

For verification of the digital signature, the server must have access to a user's certificate that's signed by a Certification Authority (CA) recognized by the server. There are several ways for the server to get access to the user's certificate:

1. The certificate is appended to the signature.
2. The public key hash is appended to the signature. The server is able to fetch the corresponding certificate from a certificate service.
3. A URL for the certificate is appended to the signature. The server is able to fetch the certificate using Internet methods.
4. The server knows the user certificate based on previous data exchange with the user (from a previous digital signature, for example).

Enhanced Security for Consumers

A number of trade organizations are working together to promote PKI:

- *The PKI Forum* operates as an autonomous, unincorporated entity within the Open Group. Founded by Baltimore Technologies, IBM, Microsoft, Entrust Technologies, and RSA Security, it is a non-profit organization committed to promoting PKI and increasing confidence in the technology.
- *Radicchio*, founded by Sonera SmartTrust with GemPlus and EDS, promotes PKI in secure, wireless e-commerce — becoming the industry voice and authority in this space. One of its stated goals is "to enable a dynamic global market for secure wireless e-commerce through high-level regulatory processes and technical collaboration and consensus between members."
- *The Mobile Electronic Signature Consortium*, formed in January 2000, is an association of companies and organizations from the mobile phone and Internet sectors. The basic pretext for forming the group is that the founder members assume the current separation of mobile telecommunications and the Internet as implemented in WAP will not last. Members are all working on the integration of mobile telecommunications and fixed-connection Internet technologies to generate services that will require a mobile digital signature as a way to establish legal security for transactions performed.

One proposal is to add a special "signature" button to mobile device keypads. This will help to create in the mind of the end user the notion that only mobile devices with such a button provide access to secure electronic/mobile commerce.

Future WAP-related Developments

The WAP Forum is working on a large number of new specifications to create a framework within which to build innovative new mobile applications. Chief of these is the so-called convergence effort: WAP version 2.0 re-specifies WML as an XHTML application in order to achieve convergence with other web content definition technologies. I'll have much more to say on this shortly.

WAP Forum Specifications

There are significant efforts by the WAP forum to develop specifications on:

- **Transport layer end-to-end security**
  A method of redirecting the client from the default pull WAP gateway to a subordinate proxy WAP gateway in the secure domain of the content server, thus allowing WTLS all the way from the client to the content server's secure domain.
- **Provisioning architecture**
  An effort to impose order and security on the process of remotely configuring connection details on subscribers' phones.
- **Wireless PKI architecture**
  This work aims to define the architecture that can provide a Public Key Infrastructure, to exploit the capabilities of mobile devices to handle private certificates, which were added to WAP in v1.2.
- **Multimedia enhancements**
  To exploit the possibilities opened up by the speed of delivery on 3G bearer networks.

Faster Phone Data Networks

The speed of bearer network technology development and implementation is just as important to the quality of the WAP user's experience as the application functionality defined in the various WAP APIs. A frequent criticism of WAP is that the reality does not live up to the hype, and at present most of us would agree with that point of view in most cases. Nonetheless, I believe that new phones supporting GPRS (or above) are crucial if WAP is to enjoy widespread acceptance and support. Other factors, such as charges and widespread availability of services are equally important.

Any new technology is only worthwhile if it's of real benefit to the consumer and makes their lives significantly easier. At present, GSM data network speeds are too slow, and there is not enough WAP content out there. However, the following bearer network technologies will be available in the not-too-distant future:

- GPRS — initial transmission speeds of 14.4 kbit/s, rising potentially to 56-100kbit/s
- HSCSD — up to 56kbit/s
- EDGE — 384kbit/s
- UMTS — theoretically 2Mbit/s

These maximum speeds may not be achieved due to network capacity limitations, but even so the fact that these connections are "always on", and the faster data transmission speeds, will be key factors in accelerating WAP acceptance.

Improved Device-to-device Communication

Another new technology that should be mentioned here is **Bluetooth**, a computing and telecommunications industry specification that describes how mobile phones, computers, and personal digital assistants (PDAs) can connect with each other (and with home and business phones and computers) using a short-range wireless connection. Bluetooth enables transmission at 1 megabit per second — five times faster than infrared transfers.

As time goes on, the distinctions between different classes of mobile devices will blur. Future devices will include elements from all of the current classes (PDAs, communicators, smartphones, and so on), and will be able to share information using Bluetooth. This creates a headache for device manufacturers, who will be expected to respond to demand either by producing a huge variety of consumer devices, each targeted at very specific audiences, or by attempting to cater for everyone at once by incorporating Bluetooth, WAP, and possibly video into the same device.

Technologies such as this allow the creation of what IBM likes to call "Pervasive" computing. This means that the network exists everywhere and every device, be it Microwave, Fridge, PC or mobile, is communicating constantly with other devices sharing data.

An Example

Innovations like these hold the promise of an exciting future for developers, as they open up the possibility of brand new types of application. For example, imagine a user listening to digital radio on their personal communicator device, and liking a particular track. The user presses the "purchase" button, and their payment details are accessed from the smart card and sent with the music vendor's details to a WAP payment service provider (PSP) over an encrypted link. The PSP handles the payment with the bank's systems, and the purchase is approved. An MP3 version of the album is then downloaded from the music vendor to the user's device.

On arriving at home, the device uses Bluetooth to transfer the album across to the user's hi-fi system, which works with their PC to get additional artwork, videos, and so on, and to make notes of links to and samples of additional material by the same or similar bands.

Does WAP Have a Future?

We've taken a look at the prospects for WAP now and in the immediate future, but it doesn't exist in a vacuum. WAP has competitors in the same space — do any of these carry a realistic threat of rendering WAP redundant?

NTT DoCoMo: I-Mode

I-mode is a hugely successful mobile Internet service in Japan. Figures released by the country's three cell phone operators recently showed that the number of Japanese mobile phone users with Internet access exceeded 10 million by the end of May (about 18% of all mobile phones in Japan), making the mobile phone Japan's most popular way of accessing the Web.

NTT DoCoMo has over 7 million subscribers, and the number is increasing by 20,000 a day. I-mode is now Japan's biggest Internet access platform. Incredibly, the popularity of the service forced DoCoMo temporarily to cap i-mode subscriptions, because of capacity problems created by the surge in subscriber numbers.

One reason for i-mode's popularity is that development is quite easy. As with WAP, sites have to be specially produced, but only a few extra HTML tags need to be learned in order to produce the CHTML (Compact HTML) content. No WAP gateway is required, so the infrastructure is simpler.

It also gives good support to retailers, as services purchased over i-mode are billed directly to the subscribers phone bill.

A WAP Competitor?

Certainly. Although NTT DoCoMo views it as a transitory technology, it's definitely an alternative to WAP, and NTT is making efforts to export the technology. It has already signed a deal with Hutchinson Whampoa in Hong Kong, and there's a possibility that i-mode will find a market here in Europe. This prospect is a very real threat to WAP since NTT DoCoMo formed an alliance with Netherlands GSM operator KPN, investing 5 billion euros and establishing a bridgehead for i-mode. Logica has just announced the first i-mode portal in the UK.

Conversely, however, i-mode itself is experiencing significant competition from WAP in Japan, as NTT's three main rivals have teamed up together to build a common WAP platform (the PacketOne system), running at 64 kbit/s over CDMA networks. Their EZWeb and Ezaccess services, which were launched in April 1999, had garnered more than a million subscribers by May 2000, based on about 210 (and growing) WAP-profiled sites.

I-mode is itself a less-robust protocol than WAP. It does not function well with packet networks and is technically weaker than WAP.

In the longer term, NTT DoCoMo is just as committed to convergence as the WAP Forum, and once this is achieved I believe that i-mode and WML will achieve equal status as XHTML applications (see below).

SIM Application Toolkit

The SIM Application Toolkit is targeted at phones that do not yet fall into the "smartphone" category. The network operator can create small programs for SIMs fairly simply. For example, SAT could be used to define how communication occurs with a bankcard that has been inserted into a dual slot phone.

Mobile banking has been the trial application with the strongest demand for SAT, but mobile e-mail and mobile information services have also been driving forces behind it.

France Telecom has just launched the second trial phase for "ItiAchat", a secure mobile e-commerce service. France leads the world on smart cards and some 35 million are in circulation. Customer's familiarity with these cards mean that a service like "ItiAchat", which is very easy to use, is likely to enjoy widespread acceptance. In this system, customers contact the merchant, by PC or telephone, and place an order. The merchant sends an SMS message back with the price to the users' Motorola StarTAC-D handset, which is equipped with a smart card reader. The consumer inserts their CB bank smart card and keys in the PIN number, upon which SMS is used to send details to the merchant and to one of the partner banks, who handle the payment. This is a neat m-commerce application, and works because consumers are comfortable with smart cards, with their bank and with the mobile phone. The client side of things is implemented on a 32K SIM toolkit card.

A WAP Competitor?

Not really. Although the SIM Application Toolkit is being heavily pushed by the smart card industry, the wider industry has already swung behind WAP — in fact, WAP 2.0 will include SAT. SAT enthusiasts talk about WAP browsers being implemented on SIM toolkit cards, but this rather ignores the fact that WAP is about rather more than just a WML browser.

Palm / AvantGo Web Clipping

In the United States the web clipping service for 3Com's Palm VII handheld device has been very successful. The Palm has a 75% market share of PDAs in this market. Sites have to be specially developed, but a number of significant content providers including AOL Instant Messenger, Amazon.com, UPS, Fedex, Yahoo! and others have developed real time content for wireless delivery via this service.

A Palm Query Application, or *PQA* as they are commonly called, is a special type of application for Palm VII devices that allows a user to interact wirelessly with web content. A PQA exploits the platform it is operating on quite well, by consolidating static parts of the application and installing that on the Palm. Links are handled locally wherever possible and only when content not held locally is referenced is an external HTTP request made.

Palm content is written in a basic form of HTML. HTML 3.2 in its entirety is not supported, only a subset — no Shockwave Flash or frames etc for the Palm VII! Curiously, the resultant set of supported tags is probably not a million miles away from the WML tag set — but without the specific support for phone-like functions.

A WAP Competitor?

Yes. 3Com will be launching Palm.net in Europe this summer, to provide 'web-clipping' of web content for the Palm VII. Web clipping may co-exist with WAP in the fragmented US market, but in Europe it is likely to be superseded, even on the Palm platform, by WAP-based services.

Recently, the Palm CEO has been quoted as saying that mobile is not a threat to Palm, but that Palm needs to identify and work with convergent technologies such as WAP, i-mode, and Bluetooth to present a major long term force.

MExE (Mobile Station Application Execution Environment)

Sun's Java 2 Micro Edition (J2ME) is a platform and an application environment. The Mobile Station Application Execution Environment (MExE) is, essentially, the incorporation of a Java virtual machine into a mobile phone. The purpose of MExE is to provide a framework on mobile phones for executing operator- or service provider-specific applications: it allows full application programming. The API integrates location services, sophisticated, intelligent customer menus, and a variety of interfaces including voice recognition. MExE will incorporate WAP, but also provides additional services exceeding the WAP functionality.

A WAP Competitor?

Potentially — this is a technology that's yet to have its day. However, it clearly needs processor capability in excess of that currently found in mobile phones, although that will not be long in coming. In many ways, MExE will be the next logical step after WAP, as the application environment is so rich.

In the end, though, I feel that it will be an additional — albeit very significant — technology found on mobile devices. Its focus remains on capabilities at the device, and consequently it is likely to run alongside WAP. It may supersede WAP as the application delivery mechanism to consumers, but it will still need to use the wireless network that WAP protocols provide.

What about Pocket PC and EPOC?

These two are worth mentioning in this context, not because they are direct competitors to WAP, but because they have a huge impact on the environment in which WAP has to operate, and consequently an indirect influence on WAP's future. They are both mobile device platforms, not solely about delivery of web-based content to mobile consumers, although of course that connectivity is central to the functionality offered to users of devices built on these platforms.

Symbian's EPOC32 is an operating system designed for small, portable computer-telephones with wireless access to phone and other information services. To earlier systems, EPOC adds wireless communication and architecture for adding application programs.

Pocket PC is the name of the latest version of Microsoft's Windows CE software (the MS operating system for small devices). Hewlett-Packard, Casio and Compaq make the hardware. Like the most advanced Palm devices, users can get e-mail, manage their calendars, and keep track of tasks. Unlike Palm, Pocket PC can also be used to read electronic books, play music, view video clips, access the Web, and download street maps directly into the organizer.

Microsoft Mobile Explorer is an HTML 3.2 compatible web browser that supports frames, JavaScript, SSL, and 128-bit encryption, but not ActiveX. It also supports WML v1.1.

Will WML be Superseded by HTML?

The argument for the possible replacement of WML by HTML is that as phone network data transmission speeds increase as a result of new technologies, the need for the 'primitive' WML language will diminish. It's certainly true that as network bandwidth increases and processors become more capable, the arguments for a compact scripting language such as WML go away. However, that's still in the future and WML fulfills a real need now. As time goes by, the numbers of different devices with different capabilities will increase — this is a good thing, as it means greater connectivity and can lead to wonderful things like voice recognition and wireless broadband. In this context, the argument of whether WML will be replaced by HTML becomes irrelevant.

Yes in some devices, HTML will be more appropriate, but in others WML, or voice, or something else, will be the browser of choice. Microsoft's Mobile Explorer, as implemented in the newly available Sony CMD-Z5 mobile, points the way towards a multi-format world by being the first dual-mode HTML and WML browser. MME also has the stated design path of future support for XML-based content. The point to take on board is that there is no single "best" solution for all mobile buyers.

For the time being, however, there isn't the network bandwidth, the device processor power, or the display capabilities to support HTML over wireless links to devices such as mobile phones.

It's likely, though, that even as data transmission speeds increase, processors become more powerful, and displays more capable, there will still be a demand for very simple communication devices. The point is that WTP and WML are optimized for their operating environments, have very wide industry support and are relatively cheap to implement in mobile devices. WAP's position seems assured, at least in the lower-end mobile phones, so WML will be the pre-eminent mobile markup language for at least the next few years. And all this time, the WAP Forum will be working to add new functionality that reflects the changes in the underlying technology, and develops capabilities that meet the special needs of mobile devices.

As time goes on, client devices will not just be restricted to PCs and mobile phones. There will be very many different devices using the Web for wide-area networking. Currently, this adds up to a headache for content providers who have to decide where to commit their scarce development resources and where to target their user base. Users too expect wide availability of web services. The WAP Forum, digital television industries and W3C recognize this, and are addressing the problem of differing client capabilities through convergence.

Convergence and Universal Accessibility

One of the weaknesses of WAP today — and of any other non-PC web access technology — is that generally, web content has to be specially authored for its target devices.

The next key focus for WAP is universal accessibility. Web content today is aimed at the de facto standard of a desktop PC with an HTML browser. WML and i-mode browsers on mobile phones are the first of new class of web clients, each with differing capabilities.

The goal of universal accessibility is to ensure access to applications and services, with proper rendering of the content regardless of client capabilities. Authors will be able to create content once, and then have it rendered as appropriate for display on different client devices.

The WAP Forum is working together with W3C to make this happen by standardizing with a new content markup language: **XHTML**.

XHTML

As the World Wide Web Consortium (W3C) describes it, XHTML (Extensible Hypertext Markup Language) is "a reformulation of HTML 4 as an application of the Extensible Markup Language (XML)".

XHTML is in fact the successor to HTML 4, which is the current standard markup language used for defining PC web content. In XHTML, all HTML 4 markup tags and attributes will continue to be supported, separated into different modules specific to different sectors of the industry. The advantages it brings are extensibility and portability.

Extensibility means that as new ideas for web communication and presentation emerge, they can be implemented without having to wait for the next major version of HTML (and browser support for it). New tags or attributes can be defined to express the new possibilities, and provided that *some* program at the receiving end can understand and act on them, new things may happen on your web page that never happened before. Specific sets of extensions for XHTML are planned for mathematical expressions, vector graphics, and multimedia applications.

WAP Version 2.0

The WAP Forum is working with W3C to converge WML 2.0 with XHTML, the intention being to ensure that WML is the wireless module of XHTML. This module contains the necessary extensions to provide backward compatibility and functionality with WML 1.x.

Content for WAP v2.0

When developing content for display on multiple client formats, it can be authored as an XML application that describes the content to be displayed, but not how to display it — that's dictated by separate stylesheets, which describe the particular style of presentation for specific devices.

Clearly, for this to work, the rendering agent must know something about the clients to which it is sending content. This is where the UAProf (and its W3C companion CC/PP) comes in. Devices publish their capabilities to the network, so that content may be rendered appropriately for their capabilities.

The specifications to put this in place are in an advanced state of readiness. Products that support this should be with us soon (some early adopters are already with us), and WAP v1.2 with UAProf firmly positions the wireless Internet world to take advantage of this.

WAP is not just WML

Within the W3C effort to develop XHTML, definition of the sector-specific modules is made the responsibility of appropriate industry groups. Hence, the WAP Forum is the industry body responsible for the wireless XHTML module.

We are rapidly approaching a more fragmented web world — a hugely increased variety of different web clients, not just the PC — and with that diversity, it becomes impossible for a single body such as W3C to oversee every aspect of web connectivity. Nor is it likely that WAP will be the single 'best' solution for all mobile devices.

Here lies the real value of the WAP Forums work. Consisting of all the industries major players as it is, it is well placed to ensure that the development of wireless connectivity meets the needs of users and the investors in the industry. Functionality specific to the needs of mobile consumers will be made available and can be utilized by mobile developers — quite possibly through WAP-WML, especially where the clients involved are cheap, mass-market mobile phones — but it could be through other markup languages.

In a converged world, it is no longer so important what the language of choice is. Developers can develop content more easily for more clients, and users find more content available to them, regardless of the client device used.

That's for the (near) future. WAP-WML has its problems, but it does effectively deliver web-based content to the kind of mobile phone devices we have available today. And it does enjoy massive support from the industry, which has already done enough to ensure that WAP browsers are going to reach the hands of a good many consumers. As Psion's Chief Technology Officer, Charles Davies told a conference audience recently, "The real point about WAP is not that it's not very good, but that it's going to be in 100 million devices."