

WML to XHTML Migration

Version 2.1; April 30, 2003

Browsing

NOKIA

Contents

1	Introduction.....	5
2	Evolution of WAP	6
2.1	Evolution of WML.....	6
3	Comparison of WML 1.x and XHTML MP and CSS	9
4	Migration Path from WML 1.x to XHTML MP	10
4.1	Developer Transition from WML to XHTML.....	10
4.2	Scalable Applications.....	10
4.3	Using Style Sheets	11
5	Nokia Solutions for Developers.....	13
5.1	Nokia Mobile Internet Toolkit	13
5.2	Nokia XHTML Browser Implementation.....	13
6	Configuring Servers to Serve XHTML/WML Content.....	14
7	Technical Details of WAP 2.0 Browser Implementation.....	15
7.1	XHTML Mobile Profile.....	15
7.2	WAP CSS	15
7.3	WML 1.x Compatibility Extensions, Informal.....	17
7.4	Key Differences between WML and XHTML Mobile Profile	17
7.5	Example Code: Linking WML and XHTML Pages.....	18
7.6	Example Style Sheet	19
8	Terms and Abbreviations	20
9	References	21

Change History

19 Nov 2001	V1.0	Initial document release
19 Nov 2001	V1.1	One reference removed
19 Nov 2001	V1.2	Minor changes
28 Nov 2001	V1.3	Minor changes
13 Mar 2002	V2.0	XHTML Mobile Profile, technical chapter, gateway and style sheet information added
30 Apr 2003	V2.1	Template changed, NDSP information added, Nokia UI categories information updated, minor updates overall, major changes to Chapter 6

Disclaimer

The information in this document is provided "as is," with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. Furthermore, information provided in this document is preliminary, and may be changed substantially prior to final release. This document is provided for informational purposes only.

Nokia Corporation disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information presented in this document. Nokia Corporation does not warrant or represent that such use will not infringe such rights.

Nokia Corporation retains the right to make changes to this specification at any time, without notice.

The phone UI images shown in this document are for illustrative purposes and do not represent any real device.

Copyright © 2003 Nokia Corporation.

Nokia and Nokia Connecting People are registered trademarks of Nokia Corporation.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc.

Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

License

A license is hereby granted to download and print a copy of this specification for personal use only. No other license to any other intellectual property rights is granted herein.

WML to XHTML Migration

Version 2.1; April 30, 2003

1 Introduction

The following document describes the evolution from Wireless Markup Language (WML) 1.x to Extensible Markup Language Mobile Profile (XHTML MP), and is intended for Wireless Application Protocol (WAP) developers who are creating mobile Internet services using WML. The goal is to give developers guidance when shifting from WML development to XHTML.

The next-generation mobile services specification is WAP 2.0, which is endorsed by Open Mobile Alliance (OMA) browsing specifications. Until WAP 2.0, there were two different specifications for languages used in the wireless and wired worlds: HTML for World Wide Web (WWW) publishing and WML for wireless services.

XHTML brings the mobile and the fixed Internet closer together; being the only specified language for both the wireless and wired Web. XHTML enables better and easier control over presentation than WML, thereby enhancing the usability of mobile services for consumers. The WAP 2.0 specification has two key elements in terms of mobile browsing: In addition to the markup language changing from WML to XHTML MP, the transmission protocol changes from WAP to Wireless Profiled TCP/IP (wTCP/IP). Both XHTML and TCP/IP are standards in the fixed Internet and this migration supports the convergence between mobile and fixed domains.

2 Evolution of WAP

WAP is a global standard for mobile Internet applications and browsing. Functionally similar to the WWW, it is designed to accommodate small devices with limited memory and small screens, as well as low bandwidth connections to the Internet that may be unreliable and have high latency. The first generation of WAP specifies an application environment including WML 1.x and a stack of optimized communication protocols.

For mobile devices, the future of WAP lies in its close alignment with widely accepted Internet standards. The former WAP Forum and the World Wide Web Consortium (W3C) have successfully defined mobile Internet standards over a period of several years. To enable the convergence of mobile and wired Web content development, the WAP Forum adopted the XHTML standard with Cascading Style Sheets (CSS) from the W3C as the basis for WAP 2.0. Since mid-2002, the WAP Forum's specification work is being continued in OMA. The transition to XHTML MP and WAP CSS strengthens the mobile browser's position within the Internet mainstream and allows for a far greater range of presentation design and formatting possibilities than previously possible.

2.1 Evolution of WML

XHTML is the foundation of the next generation of mobile browsing. The XHTML Mobile Profile markup language along with WAP CSS is most of what makes up WAP 2.0. It offers some advantages over the original WML, particularly in the area of content presentation, but it also borrows a great deal from what works well in WAP. For instance, scripting, push, and telephone functionality integration (WTAI), borrow heavily from earlier WAP specifications.

WML

WML is a markup language based on the Extensible Markup Language (XML). The official WML specification was initially developed and maintained by the WAP Forum, an industry-wide consortium founded by Nokia, Openwave, Motorola, and Ericsson.

HTML

Hypertext Markup Language (HTML) is a markup language used to create hypertext documents that are portable from one platform to another. HTML documents are SGML documents with generic semantics that are appropriate for representing information from a wide range of applications. HTML has been the *lingua franca* of WWW publishing.

XHTML MP

XHTML, according to the W3C, represents the first major change to HTML since HTML 4.0 was released in 1997. In fact, the latest version of HTML (Version 4.1) forms the basis of XHTML: all tag definitions and syntax are the same. XHTML simply adds modularity and enforces strict adherence to language rules. As a result, XHTML brings clean structure to Web pages, which is especially important given the small screens and limited power of mobile devices. W3C is recommending XHTML for all future Web development for desktops, as well as all other devices, including mobile handsets.

XHTML Basic is the mobile version of XHTML 1.1. XHTML Basic is designed for Web clients that do not support the full set of XHTML features, for example, Web clients such as mobile phones, PDAs, pagers, and set-top boxes. XHTML MP is a strict subset of XHTML, starting with XHTML Basic and adding a few elements and attributes from full XHTML 1.1 that are useful in mobile browsers including additional presentation elements and support for internal style sheets.

WAP CSS

Cascading Style Sheets (CSS) describe how documents are presented on screen in the browser. CSS separates the presentation from the content. Changes to the presentation can be made in the style sheet and the changes are automatically reflected throughout the entire document.

WAP CSS is the mobile version of CSS defined by OMA. It is a subset of CSS, omitting features that are not appropriate for very small devices and adding a few WAP-specific extensions to CSS.

WML 1.x Compatibility Extensions

The WML 2.0 specification defines WML 1.x compatibility extensions, which are used to achieve backward compatibility instead of implementing full WML 1.x functionality. These extensions make it possible to use WML 1.x specific features in WAP 2.0-compliant clients that do not have dual browsers supporting both XHTML MP and WML 1.x. Figure 1 displays the relationships between markup languages.

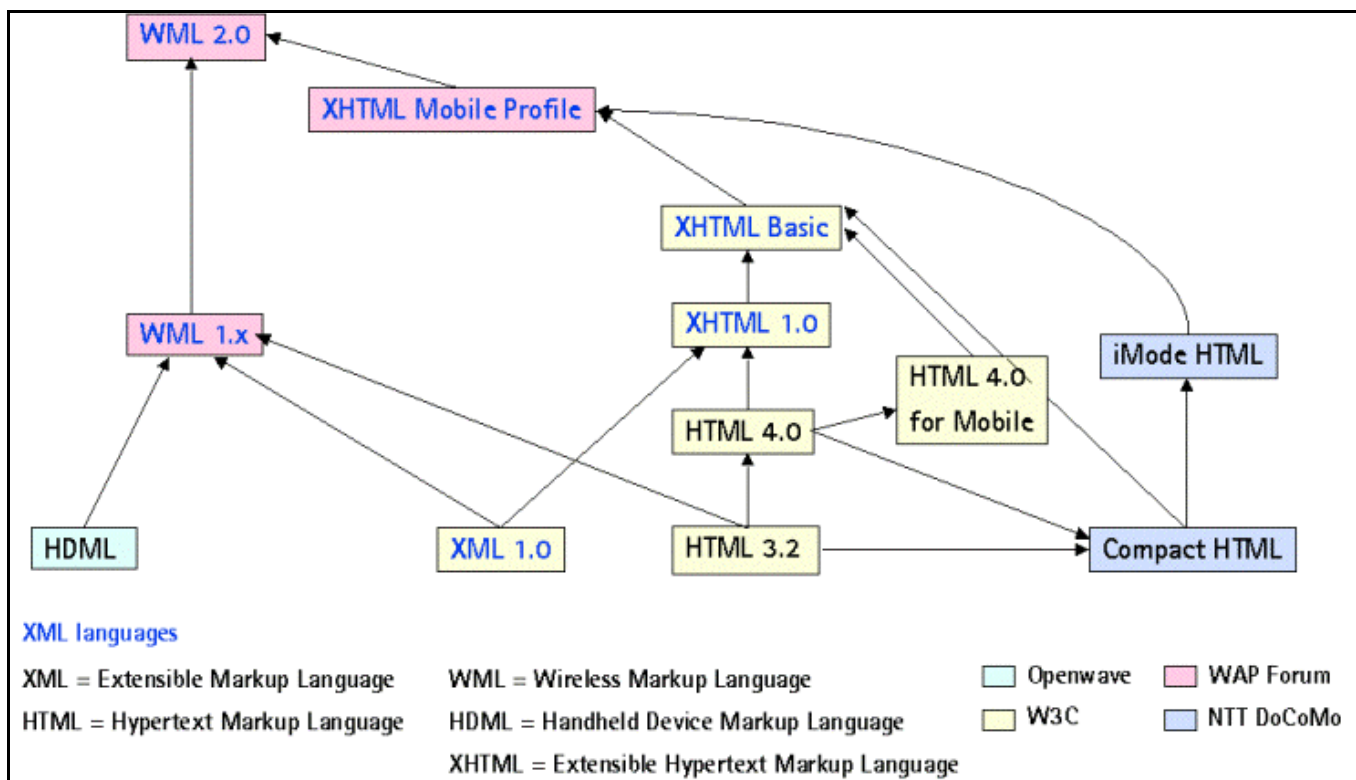


Figure 1: Relationship between markup languages

The evolution of mobile browsing is illustrated in Figure 2, which displays the most significant features that each WAP version brings to the mobile Internet.

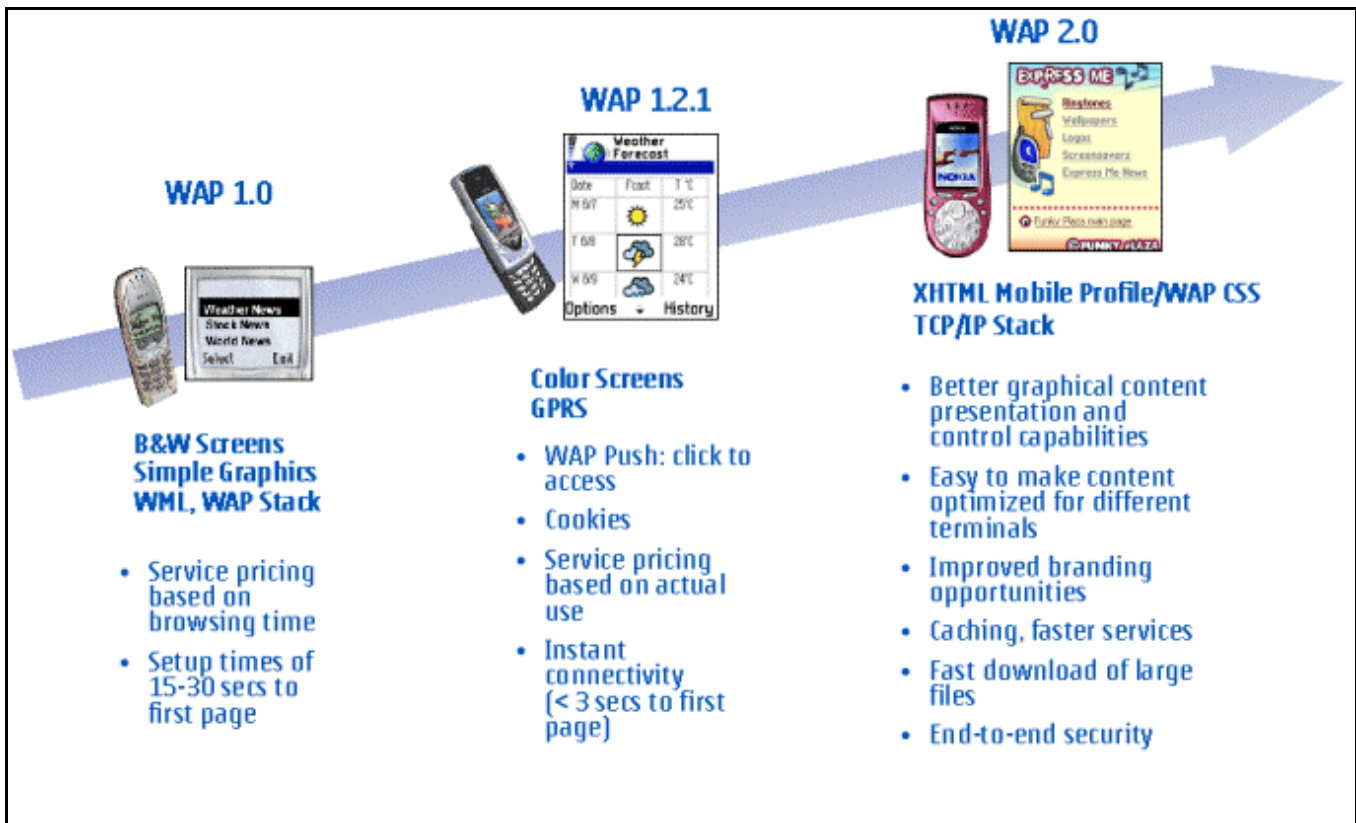


Figure 2: Evolution of mobile browsing

3 Comparison of WML 1.x and XHTML MP and CSS

Both WML 1.x and XHTML MP are optimal markup languages for devices with small screens and limited presentation capabilities, such as mobile handsets and PDAs. The languages can be used in devices with limited memory because neither the WML browser nor the XHTML MP browser requires much memory to run. Both WML and XHTML enforce strict adherence to XML language rules.

The most significant differences between WML 1.x and XHTML MP and CSS are related to content presentation capabilities and content portability across user interface categories. The transition to XHTML provides two big technical advantages compared to the existing WML environment. First, XHTML is a true industry standard supported by both the OMA and the W3C, combining the wireless and wired Web. Second, XHTML supports a wireless version of style sheets, which provides a technique for separating the content from the presentation.

Table 1 illustrates some general differences between WML 1.x and XHTML MP and CSS.

	WML 1.x	XHTML Mobile Profile and CSS
Standardization body	Standard developed by WAP Forum	Standard developed by W3C and adopted by OMA
Content displaying in devices	Content and layout defined in same document, which has to be separately tailored for each device	Content and layout defined in separate documents, thus same content renders differently using different style sheets
Content encoding	Content needs to be binary coded	No content encoding required
Document layout control	Basic	Advanced layout control with CSS
Color control support	Only color images but no color control for fonts, background, borders, etc.	Full color control support for fonts, backgrounds, borders, etc., with CSS on color devices

Table 1: General differences between WML 1.x and XHTML Mobile Profile and CSS

A good mobile service is not dependent on the technology behind the service. Despite the standard used, differences are based on smart design, a strong brand, enhanced services, and the ability to listen to the consumer's needs.

4 Migration Path from WML 1.x to XHTML MP

Until WAP 2.0, WML was used for content creation for wireless applications. From the developers' point of view, it is important to create applications that are not tied to any specific language, in anticipation of the change from WML to XHTML.

4.1 Developer Transition from WML to XHTML

Table 2 describes what a WML developer needs to do when starting to develop content for XHTML MP/CSS. It considers three different cases:

1. A developer with WML experience creating new content (the developer can use tools or primitive editors).
2. A developer with WML experience modifying existing static WML content to XHTML.
3. A developer with server-side development and WML/XML skills modifying server application to support XHTML.

	1) New XHTML Content	2) Static WML to XHTML	3) Dynamic WML to XHTML
Tools	Same	Same	Same
Code modification	Make dynamic code, data will remain unchanged when making changes	Code including data needs to be changed manually	The output when generating code needs to be changed, data remains unchanged
UI changes	With CSS	Code needs to be changed manually	With CSS

Table 2: Developer transition from WML to XHTML

In Case 1, making new XHTML code, developers can use the same tools (e.g., the Nokia Mobile Internet Toolkit) to create XHTML Mobile Profile code as they used before when making WML code. It is recommended to create dynamic XHTML code because it is easier to present the same content on various devices using style sheets. The support for CSS in XHTML Mobile Profile enables easy optimization of presentation layout for different user interfaces.

In Case 2, changing static WML to XHTML, developers can use the same tools (e.g., the Nokia Mobile Internet Toolkit) to create XHTML Mobile Profile code as they used before when making WML code. When changing static WML code to XHTML, code must be changed manually. To do UI changes, the WML code must first be changed manually to XHTML Mobile Profile, which then enables the control over presentation layout with CSS.

In Case 3, changing dynamic WML to XHTML, developers can use the same tools (e.g., the Nokia Mobile Internet Toolkit) to create XHTML Mobile Profile code as they used before when making WML code. In this case the data has already been stored in some database or, for example, XML format, and can be easily generated to XHTML Mobile Profile. With dynamic code, it is easy to define different UI styles with style sheets.

4.2 Scalable Applications

In addition to the availability of various markup languages, there is also a broad selection of terminals with different screen sizes and capabilities. Therefore, it is wise to create applications that are not tied

to any specific screen size or UI category. When designing an application, these two aspects need to be considered:

1. The independence of the markup language
2. The independence of the device's display features

The result will be applications that are scalable and rich. The biggest advantage of scalability in this case is that developers are able to generate the same content to a new markup language and display it on different devices with specific display capabilities (see Figure 3).

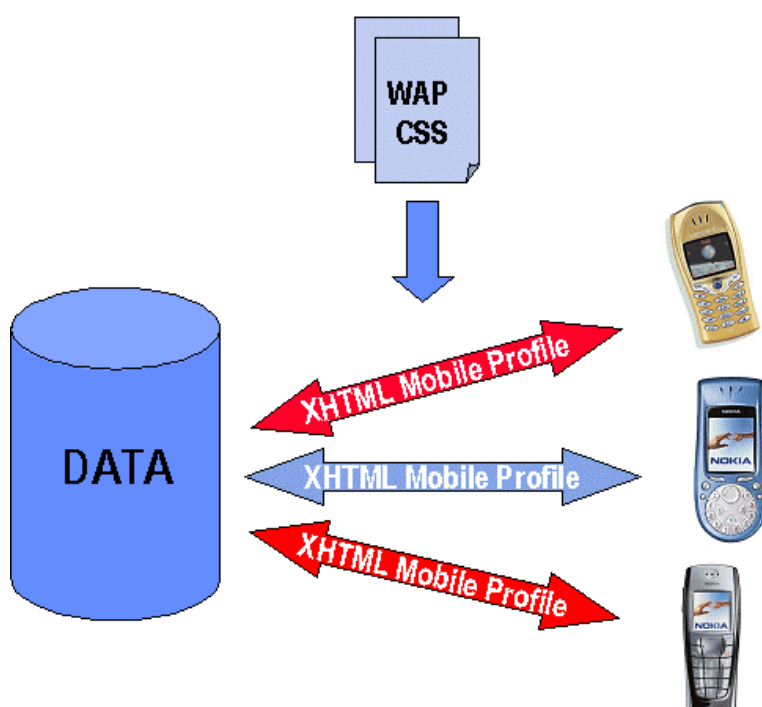


Figure 3: Same content rendered to different devices using style sheets

It is likely that most WAP applications already generate WML from data stored in some database or, for example, XML format. This is a good starting point, because it means that WAP applications can easily be adapted to a new markup language or screen size. Once a scalable application exists, it is easier to shift into generating XHTML and utilize the benefits that XHTML MP and WAP CSS bring.

4.3 Using Style Sheets

With XHTML MP and style sheets, the actual data and the presentation layer are separated from each other, which enables the use of the same content and optimized presentation for all devices by using different style sheets.

Each browser has a default style sheet, which determines how all XHTML elements are displayed. Developers can override the default style sheet by defining styles of their own. Styles can be applied in several ways. The level of priority is as follows:

1. Browser default style sheets are applied first.
2. External style sheets override default style sheets.
3. `<style>` element in document head overrides external style sheet.
4. Inline styles within a tag override any styles defined earlier for that tag only.

5. Any style with "!important" attribute overrides other definitions of that style, even if they would otherwise override (for example, "!important" can be specified to force a browser style to override an author style).

If operators want to use certain styles in their portals, developers would add a link to an external style sheet and all the content coming from developers would be rendered according to the rules in the operator style sheet.

This is how CSS work:

1. Browser fetches the first page of the application.
2. Server returns the content page (includes a "link rel" tag to style sheet).
3. Browser fetches the style sheet. A "User Agent" tag is included in the request header.
4. Server returns the appropriate style sheet for the browser type and the style sheet is cached in the browser so all subsequent pages are rendered immediately.
5. Browser fetches subsequent pages.

Even though the content for both wired and wireless Internet is authored with XHTML, it is good to remember that not all Web content becomes directly mobile just by changing the style sheet. The benefit is that the same standard can be used for creating mobile services. However, the services need to be designed and defined separately from Web applications due to the different uses and display and bandwidth differences. XHTML is only the presentation layer of the service, and good mobile applications also need working functional logic. In addition, wireless applications must be practical for mobile use.

5 Nokia Solutions for Developers

Nokia offers a wide range of tools and supporting documents for developers creating mobile applications around technologies and platforms supported by Nokia. It is expected that the XHTML MP browser and TCP/IP stack will strongly boost the use of mobile services, and browsing becomes a key enabler for various service categories.

5.1 Nokia Mobile Internet Toolkit

The Nokia Mobile Internet Toolkit is a suite of related tools that facilitate the development of applications for the mobile Internet. These tools include a set of editors for creating content of various types that are supported in mobile browsers, such as XHTML MP, WAP CSS, WML, and WMLScript. Besides offering various editors, the toolkit features debugging, logging, and testing facilities required for true application development.

In addition, the Nokia Mobile Internet Toolkit includes a handset simulator that can be used to view mobile Internet data. A device simulator is a graphical interface that simulates the display screen, keys, and controls of a mobile device. The toolkit and additional device simulators are available without charge from the Forum Nokia Web site.

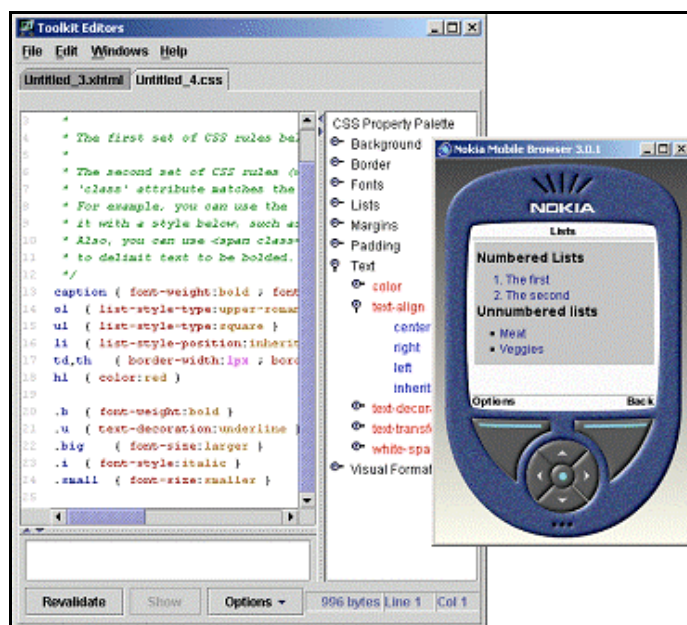


Figure 4: Nokia Mobile Internet Toolkit editors and Nokia mobile browser simulator

5.2 Nokia XHTML Browser Implementation

The first Nokia XHTML browser implementation is a dual-mode browser that natively supports both XHTML MP with WAP CSS and WML 1.x. Therefore there is no need for WML 1.x compatibility extensions when delivering the WML 1.x content to Nokia XHTML phones.

The dual-mode browser gives developers the freedom to use WML 1.x code for existing WAP applications. WML 1.x can also be used when needed for specialized WML 1.x functionality. XHTML MP with WAP CSS should be used when developers want the full expressive power available. XHTML MP and WML 1.x pages may link directly to each other using URLs in anchors, in form actions, and in other WML events. History is maintained so that the user can go back, no matter what kinds of content are intermixed. An example of creating a link from a WML deck to an XHTML page can be found in Chapter 7.

6 Configuring Servers to Serve XHTML/WML Content

XHTML MP content and WAP CSS do not require anything from WAP gateways, because there is no need for encoding. XHTML content goes through the gateway in text format, unlike WML, which is binary encoded. Nokia solves the backwards compatibility issue on the client end by having native support for both XHTML MP and WML 1.x.

WAP gateways are still required because in the first phase of evolution towards next-generation mobile browsing, XHTML enabled terminals use the WAP stack for the transport, meaning that XHTML is delivered to the terminal using the WAP stack. In phase 2, the TCP/IP stack is used, and then an update is needed to the WAP gateway to support TCP/IP. In true WAP 2.0 architecture (=XHTML MP content over TCP/IP stack) the role of the WAP gateway changes to WAP 2.0 proxy, enabling, for example, subscriber identification, service access control, privacy, charging, push services, etc.

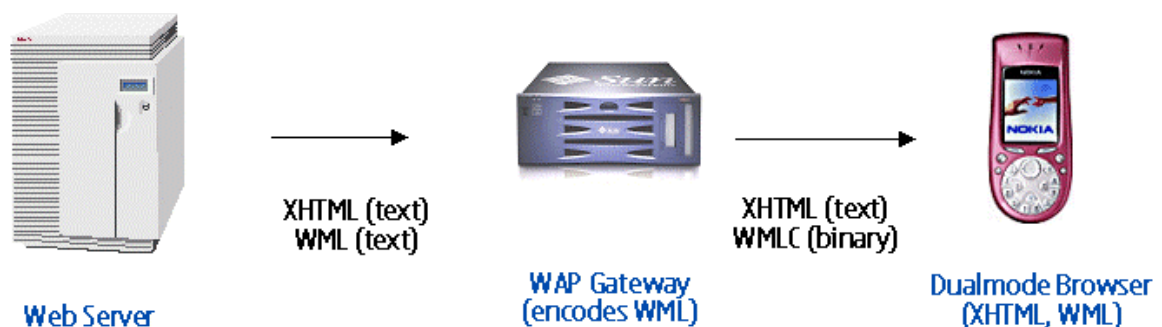


Figure 5: Delivery of XHTML MP and WML content from the Web server to the terminal via the WAP gateway

The support for WAP 2.0 introduces a few new MIME types for mobile content development. The preferred MIME type for XHTML content is "application/xhtml+xml," which should be used for serving XHTML documents to XHTML user agents. Another OMA-specified MIME type for XHTML MP content is "application/vnd.wap.xhtml+xml," but it should be noted that it may not be supported by all WAP 2.0-compliant browsers. The MIME type "text/html" is also available but, for XHTML, use of this type should be reserved for the purpose of rendering on existing HTML user agents. It should be noted that XHTML documents served as "text/html" will not be processed as XML. This means that, for example, well-formedness errors may not be detected by user agents. Authors who wish to support both XHTML and HTML user agents may utilize content negotiation by serving HTML documents as "text/html" and XHTML documents as "application/xhtml+xml".

7 Technical Details of WAP 2.0 Browser Implementation

The following chapter provides some detailed technical information and examples about WAP 2.0 and migration of WML to XHTML.

7.1 XHTML Mobile Profile

XHTML MP is the language to be used for authoring WAP 2.0 applications. XHTML Mobile Profile modules belonging to the WAP 2.0 specifications are listed in Table 3.

XHTML Basic Modules	Elements/Attributes
Structure	body, head, html, title
Text	abbr, acronym, address, blockquote, br, cite, code, dfn, div, em, h1, h2, h3, h4, h5, h6, kbd, p, pre, q, samp, span, strong, var
Hypertext	a
List	dl, dt, dd, ol, ul, li
Basic Forms	form, input, label, select, option, textarea
Basic Tables	caption, table, td, th, tr
Image	img
Object	object, param
Meta Information	meta
Link	link
Base	base
Additional XHTML Modules	Elements/Attributes
Forms (<i>partial</i>)	fieldset, optgroup
Legacy (<i>partial</i>)	start attribute on ol, value, attribute on li
Presentation (<i>partial</i>)	b, big, hr, i, small
Style Sheet	style element
Style Attribute	style attribute

Table 3: XHTML Mobile Profile modules belonging to WAP 2.0 specifications

7.2 WAP CSS

WAP CSS can be used to style mobile applications. WAP CSS properties and values belonging to the WAP 2.0 specifications are listed in Table 4.

CSS Property	Values (Optional)
border-width, border-bottom-width, border-left-width, border-right-width, border-top-width	length (0), thin, medium, thick
border-color, border-bottom-color, border-left-color, border-right-color, border-top-color	color

border-style, border-bottom-style, border-left-style, border-right-style, border-top-style	none, solid, hidden (0), dotted (0), dashed (0), double (0), groove (0), ridge (0), inset (0), outset (0)
background-color	color
background-image	uri
background-repeat	repeat, repeat-x, repeat-y, no-repeat
background-attachment	scroll, fixed
background-position	length (0), percentage (0), top, center, bottom, left, right
Clear	Left, right, both, none
Display	inline, block, list-item, none
Float	left, right, none,
Font-family	generic-family, (serif, sans-serif, monospace, cursive, fantasy), specific font family (0)
Font-size	xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, length (0), percentage (0)
Font-style	normal, italic, oblique
Font-variant	normal, small-caps
font-weight	normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900
foreground-color	color
height	length, percentage, auto
List-style-image	uri
List-style-position	inside, outside
List-style-type	disc, circle, square, decimal, upper-roman, lower-roman, lower-alpha, upper-alpha, none
Margin, margin-bottom, margin-left, margin-right, margin-top	length, percentage, auto
Padding, padding-bottom, padding-left, padding-right, padding-top	length, percentage
Text-align	left, right, center, justify (0)
Text decoration	none, underline (0), blink (0)
text-indent	length, percentage
text-transform	capitalize, uppercase, lowercase, none
Vertical-align	baseline, top, bottom, middle, sub, super
Visibility	visible, hidden, collapse (0)
-wap-marquee-dir	ltr, rtl
-wap-marquee-loop	All integers, "infinite"
-wap-marquee-style	scroll, slide, alternate
-wap-marquee-speed	slow, normal, fast
white-space	normal, pre, nowrap
width	length, percentage, auto

Table 4: WAP CSS properties and values belonging to WAP 2.0 specifications

7.3 WML 1.x Compatibility Extensions, Informal

The WML 1.x compatibility extension elements included in WML 2.0 specification are elements brought from WML 1.x because the equivalent capabilities are not found in XHTML. These extensions are needed for backwards compatibility if a full WML 1.x functionality is not implemented in the browser. These elements should not be used with the Nokia XHTML browser, since Nokia XHTML browsers have native support for WML 1.x functionality.

WML compatibility extensions use the WML namespace, identified by the "wml:" prefix. The WML 1.x compatibility extensions are listed in the Table 5, as informal note.

Element
wml:access
wml:anchor
wml:card
wml:do
wml:getvar
wml:go
wml:noop
wml:onevent
wml:postfield
wml:prev
wml:refresh
wml:setvar
wml:timer

Element	Added Attributes
body	wml:onenterbackward, wml:onenterforward, wml:ontimer, wml:newcontext
html	wml:onenterforward, wml:onenterbackward, wml:ontimer, wml:use-xml-fragments
img	wml:localsrc
input	wml:emptyok, wml:format, wml:name
meta	wml:forua
option	wml:onpick
p	wml:mode
select	wml:iname, wml:ivalue, wml:name, wml:value
textarea	wml:emptyok, wml:format, wml:name

Table 5: WML 1.x compatibility extension elements

7.4 Key Differences between WML and XHTML Mobile Profile

Most WML features are available in XHTML Mobile Profile, so WML applications can be migrated to XHTML Mobile Profile without losing significant functionality. However some features are unique to WML. The key WML features missing from XHTML Mobile Profile, along with the most equivalent XHTML feature, are listed in Table 6.

WML decks	XHTML file	XHTML substitute to the <body> is the <card> tag. There can only be a single <body> tag per page. Applications with multi-card decks need to be divided into separate pages linked together.
Variables	Form variables/server side processing	Variables are not part of XHTML Mobile Profile specification. Data sharing from one page to another could be done through an application server.
WMLScripts	Server side scripting	Without WML global variables, there is no way to share data between markup and WMLScript. WMLScripts become useless, but scripting can be done in server side.
Events	(XHTML and WCSS features)	The concept of events was unique to WML, but, e.g., "ontimer" event could be replaced with HTTP refresh.
DO element	Accesskey	Accesskey functionality is similar to WML Do elements. With access keys it is easy to navigate around the document, since Nokia handsets automatically generate a menu listing all access key attributes on the current page.

Table 6: Key differences between WML and XHTML Mobile Profile

7.5 Example Code: Linking WML and XHTML Pages

The following short code shows how to create a link from an XHTML page to a WML deck, and vice versa. A WML deck can be used, for example, to call a WMLScript file.

```
-----page1.xhtml-----

<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//WAPFORUM//DTD XHTML Mobile 1.0//EN"
"http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>XHTML Mobile Profile page</title>
</head>
<body>
<h1>A link to a WML deck</h1>
<p>
<a href="deck1.wml#go">Link</a>
</p>
</body>
</html>

-----deck1.wml-----

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
"http://www.wapforum.org/DTD/wml13.dtd">
<wml>
<card id="go" title="WML deck1">
<p>
This is a WML deck. </p>
<p>
<anchor>Here <go href="page1.xhtml"/></anchor> </p>
```

```

<p>you can go back to the page.xhtml.</p>
</card>
</wml>

```

7.6 Example Style Sheet

The example style sheet below can be used with Nokia XHTML browsers. Notice that it does not include all specified WCSS properties.

```

/*
 * A sample style sheet for XHTML Mobile Profile
 */

body {font-weight:normal; font-size:medium; font-family:
sans-serif}
p {margin-top:2px; margin-bottom:2px}
strong {font-weight:bold}
caption {text-align:center; font-weight:bold}
blink {text-decoration:blink}

li {color:black}
ol {list-style-type:decimal; margin-left:12px; margin-top:2px;
margin-bottom:2px}
ul {list-style-type:disc; margin-left:12px; margin-bottom:2px;
margin-top:2px}

table, tr {border-width:1px; border-style:solid; text-align:center}
td {align:right; valign:middle; border-width:1px; border-style:solid}
th {font-weight:bold; text-align:right}

hr {text-align:center; height:2px; width:100%; margin-top:2px;
margin-bottom:2px}

h1 {font-weight:bold; font-size:11px; margin-top:2px; margin-
bottom:2px}
h2 {font-size:11px; margin-top:2px; margin-bottom:2px; font-
weight:normal}
h3 {font-weight:bold; font-size:9px; margin-top:2px; margin-
bottom:2px}
h4 {font-size:9px; margin-top:2px; margin-bottom:2px; font-
weight:normal}
h5 {font-weight:bold; font-size:8px; margin-top:2px; margin-
bottom:2px}
h6 {font-size:8px; margin-top:2px; margin-bottom:2px; font-
weight:normal}

input {margin:1px; border-style:solid; border-width:1px; padding:2px}
textarea {border-style:solid; border-width:1px}
select {margin:1px; border-style:solid; padding:2px}

.b {font-weight:bold}
.u {text-decoration:underline}
.big {font-size:larger}
.small {font-size:smaller}

```

8 Terms and Abbreviations

Term or Abbreviation	Description
CHTML	Compact Hypertext Markup Language
CSD	Circuit Switched Data
CSS	Cascading Style Sheet
GPRS	General Packet Radio Service
HDML	Handheld Device Markup Language
HTML	Hypertext Markup Language
MIDP	Mobile Information Device Profile
MIME	Multipurpose Internet Mail Extension
MMS	Multimedia Messaging Service
Symbian OS	Symbian Operating System
UI	User Interface
WAP	Wireless Application Protocol
WAP CSS	WAP Cascading Style Sheet
WML	Wireless Markup Language
WTAI	Wireless Telephony Application Interface
XHTML	Extensible Hypertext Markup Language
XHTML MP	XHTML Mobile Profile

9 References

Browsing on Mobile Devices, <http://www.forum.nokia.com/documents>

Introduction to WAP over GPRS, <http://www.forum.nokia.com/documents>

Nokia Mobile Internet Toolkit XHTML Guidelines, <http://www.forum.nokia.com/documents>

WAP 2.0 Specifications, <http://www.wapforum.org/what/technical.htm>

XHTML Basic Specification, <http://www.w3c>. <http://www.w3.org/TR/2000/REC-xhtml-basic-20001219/>

XHTML Media Types, <http://www.w3.org/TR/xhtml-media-types/>