Winwap Technologies Oy

WinWAP Browser

Application Environment

WinWAP Browser version 4.0
WAP Specification version 2.0
Wireless Markup Language Specification version 2.0
Document dated: 26 Mar 2007

Notice of Confidentiality

Revision history

Date	Author	Description
22-Jun-2004	S Markelov	Draft version of the document.
05-Jan-2005	Maria Sandell	English spell checked.
26-Mar-2007	S Markelov	Updated and extended with XHTML and HTML static conformances.

Preamble

Wireless Application Environment (WAE) is part of the Open Mobile Alliance's effort to specify an application framework for wireless terminals such as mobile phones, pagers, and PDAs. The framework extends and leverages other WAP technologies, including WTP and WSP, as well as other Internet technologies such as XML, URLs, scripting, and various media types. The effort enables operators, manufacturers, and content developers to meet the challenges in building advanced and differentiated services and implementations in a fast and flexible manner.
WinWAP Browser is a WAE User Agent in terms of the "Wireless Application Environment Specification" Version 2.2, OMA-WAP-WAESpec-V2_2-20040609-C, Candidate Version 2.2 — 09 Jun 2004.
This document describes a set of features implemented to ensure that the WAE User Agent and the WAE Server are able to inter-operate.

Contents

1 Normative References

[HTML] "HTML 4.01 Specification", W3C, W3C Recommendation 24 December 1999.
URL: http://www.w3.org/TR/html401/

[RFC3023] "XML Media Types", M. Murata et al., January 2001.
URL: http://www.ietf.org/rfc/rfc3023.txt

[WAESpec] "Wireless Application Environment Specification — Version 2.2", Open Mobile Alliance.
OMA-WAP-WAESpec-V2_2-20040609-C.
URL: http://www.openmobilealliance.org/

[XHTMLMP] "XHTML Mobile Profile — Candidate Version 1.2", Open Mobile Alliance.
OMA-TS-XHTMLMP-V1_2-20050118-C.
URL: http://www.openmobilealliance.org/

2 WAE User Agent

2.1 Supported Media Types

WinWAP Browser supports the MIME Media Types defined in the table below. WinWAP Browser advertises a list of data types that it supports by sending MIME Media Types with the HTTP Accept header. A server may send back content with a MIME media type different from the one WinWAP Browser advertised.

Data Type	MIME Media Type	Status
WML1 textual form	text/vnd.wap.wml	✓
WML1 binary form	application/vnd.wap.wmlc	✓
WML2	application/vnd.wap.wml+xml application/wml+xml	✓
XHTML Basic	application/xhtml+xml text/html	✓
XHTML Mobile Profile	application/vnd.wap.xhtml+xml application/xhtml+xml; profile= "http://www.wapforum.org/" text/html	✓
Wireless CSS (formerly WAP CSS)	text/css	✓
WMLScript textual form	text/vnd.wap.wmlscript	✓
WMLScript binary form	application/vnd.wap.wmlscriptc	✓
ECMAScript Mobile Profile	text/ecmascript text/javascript	
WBXML	application/vnd.wap.wbxml	✓
WBMP	image/vnd.wap.wbmp	✓
VCard	text/x-vCard	
vCalendar	text/x-vCalendar	
WTA-WML	textual form text/x-wap-wta-wml	
WTA-WML binary form	application/x-wap-wta-wmlc	
Textual form of Multipart Messages that are used when the body parts are independent and need to be bundled in a particular order	multipart/mixed	
Binary form of Multipart Messages that are used when the body parts are independent and need to be bundled in a particular order	application/vnd.wap.multipart.mixed	
Textual form of Multipart Message representing objects that are aggregates of related MIME body parts	multipart/related	
Binary form of Multipart Message representing objects that are aggregates of related MIME body parts	application/vnd.wap.multipart.related	
Textual form of Multipart Message that is used when each of the body parts is alternative version of the same information	multipart/alternative	
Binary form of Multipart Message that is used when each of the body parts is alternative version of the same information	application/vnd.wap.multipart.alternative	
Textual form of Multipart Message for returning values from form	multipart/form-data	
Binary form of Multipart Message for returning values from form	application/vnd.wap.multipart.form-data	
Channels in textual form	text/vnd.wap.channel	
Channels in binary form	application/vnd.wap.channelc	
Service Indication in textual form	text/vnd.wap.si	✓
Service Indication in binary form	application/vnd.wap.sic	✓
Service Loading in textual format	text/vnd.wap.sl	✓
Service Loading in binary format	application/vnd.wap.slc	✓
Cache Operation in textual form	text/vnd.wap.co	
Cache Operation in binary form	application/vnd.wap.coc	

Data Type	MIME Media Type	Status
Provisioning Document in textual form	text/vnd.wap.connectivity-xml	
Provisioning Document in binary form	application/vnd.wap.connectivity-wbxml	

2.2 WAE Media Types

This section shows the list of WAE Media Types supported by WinWAP Browser. The “Ref.” column indicates references to the paragraphs of the [WAESpec] (p. 3). The “Status” column indicates if the feature is implemented or not in WinWAP Browser.

Item	Function	Ref.	Status
WAESpec-MT-C-001	User agent use of MIME Media type	6.1	✓
WAESpec-MT-C-002	Support for MIME Media Types	6.1	✓

Item	Function	Ref.	Status
WAESpec-ML-C-003	Support for textual form of WML2	6.2	✓
WAESpec-ML-C-004	Maintains WML context	6.2	✓
WAESpec-ML-C-005	Support for textual form of XHTMLMP	6.2	✓
WAESpec-ML-C-006	Support for the textual form of WML1	6.2	✓
WAESpec-ML-C-007	Support for the binary form of WML1	6.2	✓

Item	Function	Ref.	Status
WAESpec-STY-C-001	Allows authors to control presentation of the document	6.3	✓
WAESpec-STY-C-002	Support for Wireless CSS	6.3	✓

Item	Function	Ref.	Status
WAESpec-WMLS-C-001	Support for WMLScript execution	6.4	✓
WAESpec-WMLS-C-002	Support for WMLScript Standard Library	6.4	✓

Item	Function	Ref.	Status
WAESpec-ESMP-C-001	Support for ECMAScript Mobile Profile	6.4.4	

Item	Function	Ref.	Status
WAESpec-IMG-C-001	Support for graphical images	6.6	✓
WAESpec-IMG-C-002	Support for WBMP	6.6	✓

Item	Function	Ref.	Status
WAESpec-VCARD-C-001	Exchanges electronic business cards	6.7	
WAESpec-VCARD-C-002	Support of text/x-vCard MIME Type	6.7	
WAESpec-VCARD-C-003	vCard data port	6.7	
WAESpec-VCARD-C-004	Ability to display ‘Name’ and ‘Telephone Number’ properties	6.7	
WAESpec-VCARD-C-005	Inclusion of ‘Name’ and ‘Telephone Number’ properties	6.7	

Item	Function	Ref.	Status
WAESpec-VCAL-C-001	Exchanges calendar and scheduling information	6.8	
WAESpec-VCAL-C-002	Support of text/x-vCalendar MIME Type	6.8	
WAESpec-VCAL-C-003	vCalendar data port	6.8	
WAESpec-VCAL-C-004	Ability to display the vEvent object	6.8	

Item	Function	Ref.	Status
WAESpec-MUL-C-001	application/vnd.wap.multipart.mixed	6.9.2	
WAESpec-MUL-C-002	multipart/mixed	6.9.2	
WAESpec-MUL-C-003	application/vnd.wap.multipart.related	6.9.3	
WAESpec-MUL-C-004	multipart/related	6.9.3	
WAESpec-MUL-C-005	application/vnd.wap.multipart.alternative	6.9.4	
WAESpec-MUL-C-006	multipart/alternative	6.9.4	
WAESpec-MUL-C-007	Support for the multipart message returning a set of values from a form	6.9.5	
WAESpec-MUL-C-008	application/vnd.wap.multipart.form-data	6.9.5	
WAESpec-MUL-C-009	multipart/form-data	6.9.5	

2.3 WAE Features

WAE User Agents do not depend on any particular transport protocol, although WAE only defines a browser model of User Agent. User Agents are required to provide a hypermedia transfer service. The combination of WSP (Wireless Session Protocol) and WTP (Wireless Transaction Protocol) provides the hypermedia transfer service over secure and non-secure datagram transports. The HTTP (Hypertext Transfer Protocol) provides the hypermedia transfer service over secure and non-secure connection-oriented transports. WAE User Agents must, at a minimum, implement either WSP or Wireless Profiled HTTP. The network communication takes place in the form of WSP/HTTP 1.1 headers and content.

The “Ref.” column indicates references to the paragraphs of the [WAESpec] (p. 3).

Item	Function	Ref.	Status
WAESpec-HTS-C-001	Support for Hypermedia Transfer Service	7.1.1	✓
WAESpec-HTS-C-002	Support for WSP	7.1.1	✓
WAESpec-HTS-C-003	Support for W-HTTP	7.1.1	✓
WAESpec-HTS-C-004	Support for Caching Model	7.1.2	✓

Item	Function	Ref.	Status
WAESpec-URI-C-001	Minimum URI length	7.2	✓
WAESpec-URI-C-002	HTTP URL Scheme	7.2.1	✓
WAESpec-URI-C-003	HTTPS URL Scheme	7.2.2	✓
WAESpec-URI-C-004	HTTPS URI Scheme over W-HTTP	7.2.2	✓
WAESpec-URI-C-005	HTTPS URI Scheme over WSP	7.2.2	✓
WAESpec-URI-C-006	Report an error when no TLS or WTLS support available	7.2.2	✓

Item	Function	Ref.	Status
WAESpec-DL-C-001	Support for OMA Download	7.1.5	
WAESpec-DRM-C-001	Support for OMA DRM	7.3.2	

Item	Function	Ref.	Status
WAESpec-PUSH-C-001	Support for Push	7.5.2	✓
WAESpec-PUSH-C-002	application/vnd.wap.multipart.mixed	7.5.2	
WAESpec-PUSH-C-003	multipart/mixed	7.5.2	
WAESpec-PUSH-C-004	application/vnd.wap.multipart.related	7.5.2	
WAESpec-PUSH-C-005	multipart/related	7.5.2	
WAESpec-PUSH-C-006	application/vnd.wap.multipart.alternative	7.5.2	
WAESpec-PUSH-C-007	multipart/alternative	7.5.2	
WAESpec-PUSH-C-008	undefined push behaviour	7.5.2	✓
WAESpec-PUSH-C-009	Support for Push Message	7.5.2	✓
WAESpec-PUSH-C-010	Support for Push OTA	7.5.2	
WAESpec-PUSH-C-011	Support for Service Indication	7.5.2	✓

Item	Function	Ref.	Status
WAESpec-MMS-C-001	Support for MMS	7.4	

Item	Function	Ref.	Status
WAESpec-I18N-C-001	Support for UTF-8	7.6.1	✓

Item	Function	Ref.	Status
WAESpec-I18N-C-002	Support for UTF-16	7.6.1	
WAESpec-I18N-C-003	Treat the character encoding of an XML document as defined in [RFC3023].	7.6.1	✓

Item	Function	Ref.	Status
WAESpec-PICT-C-001	Support for Pictograms	7.6.2	

Item	Function	Ref.	Status
WAESpec-UAC-C-001	Informs supported media type using Accept header	7.7.1	✓
WAESpec-UAC-C-002	Informs supported character encoding using Accept-Charset header	7.7.1	✓
WAESpec-UAC-C-003	Informs supported contentcodings using Accept-Encoding header	7.7.1	✓
WAESpec-UAC-C-004	Informs supported language using Accept-Language header	7.7.1	

Item	Function	Ref.	Status
WAESpec-EFI-C-001	Support for EFI	7.10.1	

Item	Function	Ref.	Status
WAESpec-PSTOR-C-001	Support for Persistent Storage	7.12.3	
WAESpec-PSTOR-C-002	Managing access to store objects	7.12.3	

Item	Function	Ref.	Status
WAESpec-SEC-C-001	Support for HTTP/1.1 Basic Authentication	7.12.1	✓
WAESpec-SEC-C-002	Support for WMLScript Crypto Library	7.12.3	
WAESpec-SEC-C-003	Support for ESMP Crypto	7.12.5	

2.4 User Agent Behaviour

The WAE User Agent includes a navigational history model that allows the author to manage backward navigation in a convenient and efficient manner. The user agent history is modelled as a stack of entries that represent the resources in the navigational path the user traversed to arrive at the current location. The stack is configured temporally, such that the newest entry is at the top of the stack and the oldest entry is at the bottom of the stack.

The "Ref." column indicates references to the paragraphs of the [WAESpec] (p. 3).

Item	Function	Ref.	Status
WAESpec-UAB-C-001	Navigation History	7.13.1	✓
WAESpec-UAB-C-002	Access to Back key at all times	7.13.2	✓
WAESpec-UAB-C-003	User Agent performs pop operation on BACK	7.13.2	✓
WAESpec-UAB-C-004	WML1 User Agent executes `prev` task on BACK	7.13.2	✓
WAESpec-UAB-C-005	WML1 `do type=prev` behaviour	7.13.2	✓

Item	Function	Ref.	Status
WAESpec-WTA-C-001	Support for WTAI Public URI functions	7.14.1	✓[1]
WAESpec-WTA-C-002	Support for WTAI Public WMLScript functions	7.14.1	

[1] Turning a dial only

3 XHTML Mobile Profile Static Conformances

3.1 The XHTML Mobile Profile 1.2 Document Type

The XHTML Mobile Profile 1.2 document type is an XHTML document type based upon the module framework and the modules. The XHTML Mobile Profile 1.2 document type is defined as a strict superset of XHTML Basic. The XHTML Mobile Profile 1.2 document type is a strict superset of XHTML Mobile Profile 1.1.

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

3.1.1 XHTML Basic Modules

Item	Function	Ref.	Status
XHTMLMP-XHTMLMOD-C-001	XHTML Structure module	5	✓
XHTMLMP-XHTMLMOD-C-002	XHTML Text module	5	✓
XHTMLMP-XHTMLMOD-C-003	XHTML Hypertext module	5	✓
XHTMLMP-XHTMLMOD-C-004	XHTML List module	5	✓
XHTMLMP-XHTMLMOD-C-005	XHTML Basic Forms module	5	✓
XHTMLMP-XHTMLMOD-C-006	XHTML Basic Tables module	5	✓
XHTMLMP-XHTMLMOD-C-007	XHTML Image module	5	✓
XHTMLMP-XHTMLMOD-C-008	XHTML Object module	5	
XHTMLMP-XHTMLMOD-C-009	XHTML Metainformation module	5	✓
XHTMLMP-XHTMLMOD-C-010	XHTML Link module	5	✓
XHTMLMP-XHTMLMOD-C-011	XHTML Base module	5	✓

3.1.2 Other XHTML Modules

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-XHTMLMOD-C-012	XHTML Style Sheet module	5	✓
XHTMLMP-XHTMLMOD-C-013	XHTML Style Attribute module	5	✓
XHTMLMP-XHTMLMOD-C-023	XHTML Intrinsic Events module	5	
XHTMLMP-XHTMLMOD-C-024	XHTML Scripting module	5	

3.1.3 Other XHTML Elements and Attributes

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-XHTMLMOD-C-014	`fieldset` element in Forms module	5	✓
XHTMLMP-XHTMLMOD-C-015	`optgroup` element in Forms module	5	✓
XHTMLMP-XHTMLMOD-C-016	`start` attribute on `ol`	5	✓
XHTMLMP-XHTMLMOD-C-017	`value` attribute on `li`	5	✓
XHTMLMP-XHTMLMOD-C-018	`b` element in Presentation module	5	✓
XHTMLMP-XHTMLMOD-C-019	`big` element in Presentation module	5	✓
XHTMLMP-XHTMLMOD-C-020	`hr` element in Presentation module	5	✓

Item	Function	Ref.	Status
XHTMLMP-XHTMLMOD-C-021	`i` element in Presentation module	5	✓
XHTMLMP-XHTMLMOD-C-022	`small` element in Presentation module	5	✓

3.2 User Agent Conformance

WinWAP Browser meets all the user agent conformance requirements. WinWAP Browser accepts XHTML Mobile Profile documents identified as “application/vnd.wap.xhtml+xml”, “application/vnd.wap.xhtml+xml” or “text/html”. As there are no conformance rules for documents with type “text/html”, there is no easy way to determine which documents of type “text/html” are XHTML Mobile Profile documents, except that the document may include the DOCTYPE declaration. WinWAP Browser uses the following HTTP header fields to declare support for XHTML Mobile Profile:

```
Accept: application/xhtml+xml; profile="http://www.wapforum.org/xhtml"
Accept: application/vnd.wap.xhtml+xml
```

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

3.2.1 XHTML User Agent Conformance

Item	Function	Ref.	Status
XHTMLMP-XHTMLUA-C-001	“Modularization of XHTML” conformance requirements	7.2	✓

3.2.2 Document Types

Item	Function	Ref.	Status
XHTMLMP-DOC-C-001	Accept XHTML Mobile Profile documents	7.2	✓
XHTMLMP-DOC-C-002	Advertise support for XHTML Mobile Profile documents	7.2	✓

3.3 Style Sheets

WinWAP Browser supports the use of style to provide authors control of presentation. WinWAP Browser supports style and style language shall be WCSS.

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-STYLE-C-001	Support for WAP CSS	8	✓
XHTMLMP-STYLE-C-002	Handling of type “text/css” for external style sheet	8.1.1	✓
XHTMLMP-STYLE-C-003	Handling of type “text/css” for internal style sheet	8.1.2	✓
XHTMLMP-STYLE-C-004	Default type “text/css” for inline style rules	8.1.3	✓

See the “WinWAP Browser, WAP Cascading Style Sheets” for detailed descriptions.

3.4 Scripting

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-SCRIPT-C-001	Support for scripting		
XHTMLMP-SCRIPT-C-002	Support for ESMP		
XHTMLMP-SCRIPT-C-003	Script reference processing model	9.2.1	
XHTMLMP-SCRIPT-C-004	Processing `script` element	9.3.1.1	
XHTMLMP-SCRIPT-C-005	Processing `noscript` element	9.3.1.2	
XHTMLMP-SCRIPT-C-006	Accepts ESMP with MIME media type text/ecmascript	9.2.1.3	
XHTMLMP-SCRIPT-C-007	Accepts ESMP with MIME media type text/javascript	9.2.1.3	
XHTMLMP-SCRIPT-C-008	Support for other scripting languages	9.1	

3.5 Events

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-EVENT-C-001	Support for XHTML Mobile Profile event model	10.1	
XHTMLMP-EVENT-C-002	Load event	10.3.1	
XHTMLMP-EVENT-C-003	Support for Load event on body element	10.3.1	
XHTMLMP-EVENT-C-004	Unload event	10.3.2	
XHTMLMP-EVENT-C-005	Support for Unload event on body element	10.3.2	
XHTMLMP-EVENT-C-006	Click event	10.3.3	
XHTMLMP-EVENT-C-007	Support for Click event on mandatory elements	10.3.3	
XHTMLMP-EVENT-C-008	Support for Click event on all specified element	10.3.3	
XHTMLMP-EVENT-C-009	Double Click event	10.3.4	
XHTMLMP-EVENT-C-010	Support for Double Click event on mandatory elements	10.3.4	
XHTMLMP-EVENT-C-011	Support for Double Click event on all specified elements	10.3.4	
XHTMLMP-EVENT-C-012	Mouse Down event	10.3.14	
XHTMLMP-EVENT-C-013	Support for Mouse Down event on all specified elements	10.3.14	
XHTMLMP-EVENT-C-014	Mouse Up	10.3.14	
XHTMLMP-EVENT-C-015	Support for Mouse Up event on all specified elements	10.3.14	
XHTMLMP-EVENT-C-016	Mouse Over event	10.3.14	
XHTMLMP-EVENT-C-017	Support for Mouse Over event on all specified elements	10.3.14	
XHTMLMP-EVENT-C-018	Mouse Move event	10.3.14	
XHTMLMP-EVENT-C-019	Support for Mouse Move event on all specified elements	10.3.14	
XHTMLMP-EVENT-C-020	Mouse Out event	10.3.14	
XHTMLMP-EVENT-C-021	Support for Mouse Out event on all specified elements	10.3.14	
XHTMLMP-EVENT-C-022	Focus event	10.3.5	
XHTMLMP-EVENT-C-023	Support for Focus event on mandatory elements	10.3.5	
XHTMLMP-EVENT-C-024	Blur event	10.3.6	
XHTMLMP-EVENT-C-025	Support for Blur event on mandatory elements	10.3.6	
XHTMLMP-EVENT-C-026	Key Press event	10.3.7	
XHTMLMP-EVENT-C-027	Support for Key Press event on mandatory elements	10.3.7	
XHTMLMP-EVENT-C-028	Support for Key Press event on all specified elements	10.3.7	
XHTMLMP-EVENT-C-029	Key Down event	10.3.8	
XHTMLMP-EVENT-C-030	Support for Key Down event on mandatory elements	10.3.8	
XHTMLMP-EVENT-C-031	Support for Key Down event on all specified elements	10.3.8	

Item	Function	Ref.	Status
XHTMLMP-EVENT-C-032	Key Up event	10.3.9	
XHTMLMP-EVENT-C-033	Support for Key Up event on mandatory elements	10.3.9	
XHTMLMP-EVENT-C-034	Support for Key Up event on all specified elements	10.3.9	
XHTMLMP-EVENT-C-035	Submit event	10.3.10	
XHTMLMP-EVENT-C-036	Support for Submit event on form element	10.3.10	
XHTMLMP-EVENT-C-037	Reset event	10.3.11	
XHTMLMP-EVENT-C-038	Support for Reset event on form element	10.3.11	
XHTMLMP-EVENT-C-039	Select event	10.3.12	
XHTMLMP-EVENT-C-040	Support for Select event on input, textarea elements	10.3.12	
XHTMLMP-EVENT-C-041	Change event	10.3.13	
XHTMLMP-EVENT-C-042	Support for Change event on input, select, textarea elements	10.3.13	
XHTMLMP-EVENT-C-043	Support for single event handler per element per event	10.4	
XHTMLMP-EVENT-C-044	Registration	10.4	
XHTMLMP-EVENT-C-045	Attempt to modify the value causes deregistration	10.4	
XHTMLMP-EVENT-C-046	Support for cancellable events	10.5	

3.6 The object Element

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-OBJECT-C-001	Support for object element	11.1	
XHTMLMP-OBJECT-C-002	Conform to generic rules for rendering the object element ([HTML4], section 13.3)	11.1	
XHTMLMP-OBJECT-C-003	Support for ‘data’ attribute as the location of the object’s data	11.2	
XHTMLMP-OBJECT-C-004	Support for ‘classid’ attribute as the location of the object’s implementation	11.3	
XHTMLMP-OBJECT-C-005	The ‘classid’ attribute takes precedence over ‘data’ attribute when both are specified	11.1	
XHTMLMP-OBJECT-C-006	Pass parameters defined with `param` element to local app	11.2, 11.3	
XHTMLMP-OBJECT-C-007	Use of PUSH Application ID to identify the location application	11.3	
XHTMLMP-OBJECT-C-008	Support for declared objects (‘declare’ attribute)	11.4	

3.7 Navigation Optimizations

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-NAVOPT-C-001	Presentation hint indicating presence of an access key for an element	12.1	
XHTMLMP-NAVOPT-C-002	Use of the ‘title’ attribute of the anchor element as the link title when presenting page’s links in a navigation menu	12.2	
XHTMLMP-NAVOPT-C-003	Provide user access to linked resources defined by the link element, e. g. through a navigation menu	12.3	
XHTMLMP-NAVOPT-C-004	Preloading of linked resources	12.3	

3.8 Text Input Modes

The “Ref.” column indicates references to the paragraphs of the [XHTMLMP] (p. 3).

Item	Function	Ref.	Status
XHTMLMP-INPUTMODE-C-001	Support for Text Input Modes module	13	
XHTMLMP-INPUTMODE-C-002	Support for ‘inputmode’ attribute on `input` element	13.2	
XHTMLMP-INPUTMODE-C-003	Support for ‘inputmode’ attribute on `textarea` element	13.2	
XHTMLMP-INPUTMODE-C-004	Support for TextInputModes attribute data type	13.2	

4 HTML Static Conformances

4.1 HTML Document Representation

Function	Ref.	Status
The Document Character Set	5.1	✓
Choosing an encoding	5.2.1	✓
Specifying the character encoding	5.2.2	✓
Numeric character references	5.3.1	✓
Character entity references	5.3.2	✓
Undisplayable characters	5.4	✓

4.2 The global structure of an HTML document

Function	Ref.	Status
The HTML element	7.3	✓
The HEAD element	7.4.1	✓
The TITLE element	7.4.2	✓
The title attribute	7.4.3	✓
Meta data	7.4.4	✓
The BODY element	7.5.1	✓
Element identifiers: the id and class attributes	7.5.2	✓
Block-level and inline elements	7.5.3	✓
Grouping elements: the DIV and SPAN elements	7.5.4	
Headings: The H1, H2, H3, H4, H5, H6 elements	7.5.5	✓
The ADDRESS element	7.5.6	

4.3 Language information and text direction

Function	Ref.	Status
Specifying the language of content: the lang attribute	8.1	
Specifying the direction of text and tables: the dir attribute	8.2	

4.4 Text

The ✓* sign indicates partially supported item.

Function	Ref.	Status
White space	9.1	✓
Phrase elements: EM, STRONG, DFN, CODE, SAMP, KBD, VAR, CITE, ABBR, and ACRONYM	8.2	✓*
Quotations: The BLOCKQUOTE and Q elements	9.2.2	
Subscripts and superscripts: the SUB and SUP elements	9.2.3	
Paragraphs: the P element	9.3.1	✓
Controlling line breaks	9.3.2	✓
Preformatted text: The PRE element	9.3.4	✓
Visual rendering of paragraphs	9.3.5	✓

Function	Ref.	Status
Marking document changes: The INS and DEL elements	9.4	

4.5 Lists

Function	Ref.	Status
Unordered lists (UL), ordered lists (OL), and list items (LI)	10.2	✓
Definition lists: the DL, DT, and DD elements	10.3	✓
The DIR and MENU elements	10.4	

4.6 Tables

The ✓* sign indicates partially supported item.

Function	Ref.	Status
The TABLE element	11.2.1	✓
Table Captions: The CAPTION element	11.2.2	✓
Row groups: the THEAD, TFOOT, and TBODY elements	11.2.3	
Column groups: the COLGROUP and COL elements	11.2.4	
Table rows: The TR element	11.2.5	✓
Table cells: The TH and TD elements	11.2.6	✓*
Borders and rules	11.3.1	
Horizontal and vertical alignment	11.3.3	✓
Cell margins	11.3.3	
Categorizing cells	11.4.2	
Algorithm to find heading information	11.4.3	

4.7 Links

The ✓* sign indicates partially supported item.

Function	Ref.	Status
Visiting a linked resource	12.1.1	✓
Other link relationships	12.1.2	
Specifying anchors and links	12.1.3	
Link titles	12.1.4	
Internationalization and links	12.1.5	
The A element	12.2	✓
Syntax of anchor names	12.2.1	
Nested links are illegal	12.2.2	✓
Anchors with the id attribute	12.2.3	
Unavailable and unidentifiable resources	12.2.4	✓
Document relationships: the LINK element	12.3	
Forward and reverse links	12.3.1	
Links and external style sheets	12.3.2	✓
Links and search engines	12.3.3	
Path information: the BASE element	12.4	✓

Function	Ref.	Status
Resolving relative URIs	12.4.1	✓
Table Captions: The CAPTION element	11.2.2	✓
Row groups: the THEAD, TFOOT, and TBODY elements	11.2.3	
Column groups: the COLGROUP and COL elements	11.2.4	
Table rows: The TR element	11.2.5	✓
Table cells: The TH and TD elements	11.2.6	✓*
Borders and rules	11.3.1	
Horizontal and vertical alignment	11.3.3	✓
Cell margins	11.3.3	
Categorizing cells	11.4.2	
Algorithm to find heading information	11.4.3	

4.8 Objects, Images, and Applets

Function	Ref.	Status
Including an image: the IMG element	13.2	✓
Generic inclusion: the OBJECT element	13.3	
Including an applet: the APPLET element	13.4	
Image maps	13.6	

4.9 Style Sheets

Function	Ref.	Status
Setting the default style sheet language	14.2.1	
Inline style information	14.2.2	✓
Header style information: the STYLE element	14.2.3	✓
Media types	14.2.4	
Preferred and alternate style sheets	14.3.1	
Specifying external style sheets	14.3.1	✓
Media-dependent cascades	14.4.1	
Inheritance and cascading	14.4.2	✓

4.10 Alignment, font styles, and horizontal rules

The ✓* sign indicates partially supported item.

Function	Ref.	Status
Background color	15.1.1	✓
Alignment	15.1.2	✓
Floating objects	15.1.3	✓*
Font style elements: the TT, I, B, BIG, SMALL, STRIKE, S, and U elements	15.2.1	✓*
Rules: the HR element	15.3	✓
Media types	14.2.4	
Preferred and alternate style sheets	14.3.1	
Specifying external style sheets	14.3.1	✓
Media-dependent cascades	14.4.1	

Function	Ref.	Status
Inheritance and cascading	14.4.2	✓

4.11 Frames

Function	Ref.	Status
The FRAMESET element	16.2.1	
The FRAME element	16.2.2	
Setting the default target for links	16.3.1	
Target semantics	16.3.2	
The NOFRAMES element	16.4.1	
Long descriptions of frames	16.4.2	
Inline frames: the IFRAME element	16.5	

4.12 Forms

The ✓* sign indicates partially supported item.

Function	Ref.	Status
The FORM element	17.3	✓
Control types created with INPUT	17.4.1	✓*
The BUTTON element	17.5	
The SELECT, OPTGROUP, and OPTION elements	17.6	✓
The TEXTAREA element	17.7	✓
The LABEL element	17.9.1	
Adding structure to forms: the FIELDSET and LEGEND elements	17.10	
Tabbing navigation	17.11.1	✓
Access keys	17.11.2	
Disabled controls	17.12.1	✓
Read-only controls	17.12.2	✓
Form submission method	17.13.1	✓

4.13 Scripts

Function	Ref.	Status
The SCRIPT element	18.2.1	
Specifying the scripting language	18.2.2	
Intrinsic events	18.2.3	
Dynamic modification of documents	18.2.4	
The NOSCRIPT element	18.3.1	