XHTML 1.1 represents a major step by the W3C in moving away from the often-disorderly world of HTML to the organized and orderly world of XML. In XHTML 1.1, the elements and attributes, which were once listed as *deprecated,* have been dropped altogether. Gone also is the more forgiving *Transitional DTD*. Strict compliance and conformity to the standard are now the order of the day. However, in exchange, XHTML now can be extended in a way that HTML couldn't. With a little bit of expertise in XML, you can now create your own elements and attributes or extend existing modules.

The tables that follow are taken from the W3C's XHTML Modularization page and list the attribute types, data types, and the modules that make up module-based XHTML.

TIP

You can find the complete specification at the following URL: www.w3.org/TR/xhtml-modularization/abstract_modules.html#s_xhtmlmodules.

Attribute Groups, Value Types, and Data Types

In order to understand the charts of the various modules, you will find it helpful to review the attribute groups, attribute types, and data types developed by the W3C. Otherwise, terms such as *PCDATA*, *Charset*, *MediaType*, and so on may not make much sense. The following charts, taken from the W3C's XHTML Modularization page, list the attribute groups and data types for you.

Attribute Groups

Rather than list all the applicable attributes for each element, the W3C has compiled them into groups or collections. When you see an element in the module charts, the chart will list both the specific or unique attributes that apply to that element, as well as the attribute groups that can be used with that element.

Attribute Group	Contains These Attributes
`Core`	class, id, title
`Events`	onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress, onkeydown, onkeyup
`Style`	style
`I18N`	xml:lang
`Common`	Core + Events + Style + I18N

Value Types

Attributes often require a particular type of value. These values and their descriptions are listed in the following table:

Value Type	Explanation
CDATA	Character data
ID	A document-unique identifier
IDREF	A reference to a document-unique identifier
IDREFS	A space-separated list of references to document unique identifiers
NAME	A name with the same character constraints as ID above
NMTOKEN	A name composed of only name tokens as defined in XML 1.0
NMTOKENS	One or more space-separated NMTOKEN values
PCDATA	Processed character data

Data Types

Occasionally, you will need to enter a particular kind of data as an attribute value. These data types and their descriptions are listed in the following table:

Data Type	Explanation
Character	A single character
Charset	A character encoding
Charsets	A space-separated list of character encodings
Color	Either a hexadecimal number (prefixed by a hash mark) or one of the sixteen color names
ContentType	A media type
ContentTypes	A comma-separated list of media types
Coords	A comma-separated list of coordinates to use in defining areas
Datetime	Date and time information
FPI	A character string representing an SGML Formal Public Identifier
FrameTarget	Frame name used as destination for results of certain actions
LanguageCode	A language code
Length	A value in pixels or percentage of the available horizontal or vertical space
LinkTypes	A space-separated list of link types, including Alternate, Stylesheet, Start, Next, Prev, Contents, Index, Glossary, Copyright, Chapter, Section, Subsection, Appendix, Help, Bookmark
MediaDesc	A comma-separated list of media descriptors, including screen, tty, tv, projection, handheld, print, braille, aural, all
MultiLength	A length or a relative length
MultiLengths	A comma separated list of items of the type, MultiLength

Data Type	Explanation
Number	One or more digits
Pixels	An integer representing the number of pixels on the screen (or paper)
Script	The content of the <script> element and the value of intrinsic event attributes
Shape	The shape of a region
Text	Textual data, generally intended to be human-readable
URI	A Uniform Resource Identifier (formerly referred to as URL, Uniform Resource Locator)
URIs	A space-separated list of Uniform Resource Identifiers

XHTML 1.1 Modules

Even if you are not particularly interested in writing strict XHTML 1.1 documents at this point, you are likely to find the concept of XHTML modules very helpful in learning how to work with XHTML markup. In the past, HTML charts generally listed elements alphabetically, leaving you to search through a long list to find the element you need. Then trying to figure out which elements and attributes worked together was an adventure all its own. With modules, elements that have related functions are grouped together. Also, attributes have been collected into easy-to-understand groups as well.

XHTML Core Modules

The following tables list the basic or "core" XHTML modules. These modules must be present in any document type that is an XHTML document. The core modules include the Structure, Text, Hypertext, and List modules.

The Structure Module

The Structure module contains the elements that define the major portions of an XHTML document. This module contains the following elements and attributes:

Elements	Attributes	Minimal Content Model
body	common	(heading | block | list)
head	I18N, profile(URI)	title
html	I18N, version, xmlns (URI="http://www.w3.org/1999/xhtml")	head, body
title	I18N	PCDATA

The Text Module

The Text module contains the basic elements used for working with text. Use the elements of this module to define headings, paragraphs, and so on. The following table lists the elements and attributes for the Text module:

Element	Attributes	Minimal Content Model
`abbr`	Common	(PCDATA | Inline)
`acronym`	Common	(PCDATA | Inline)
`address`	Common	(PCDATA | Inline)
`blockquote`	Common, cite (URI)	(PCDATA | Heading | Block | List)
`br`	Core	Empty
`cite`	Common	(PCDATA | Inline)
`code`	Common	(PCDATA | Inline)
`dfn`	Common	(PCDATA | Inline)
`div`	Common	(PCDATA | Flow)
`em`	Common	(PCDATA | Inline)
`h1`	Common	(PCDATA | Inline)
`h2`	Common	(PCDATA | Inline)
`h3`	Common	(PCDATA | Inline)
`h4`	Common	(PCDATA | Inline)
`h5`	Common	(PCDATA | Inline)
`h6`	Common	(PCDATA | Inline)
`kbd`	Common	(PCDATA | Inline)
`p`	Common	(PCDATA | Inline)
`pre`	Common xml:`space="preserve"`	(PCDATA | Inline)
`q`	Common, cite (URI)	(PCDATA | Inline)
`samp`	Common	(PCDATA | Inline)
`span`	Common	(PCDATA | Inline)
`strong`	Common	(PCDATA | Inline)
`var`	Common	(PCDATA | Inline)

The Hypertext Module

The Hypertext module contains the element necessary for creating hyperlinks between documents. The following table contains the Hypertext module:

Elements	Attributes	Minimal Content Model
a	Common, accesskey (Character), charset (Charset), href (URI), hreflang (LanguageCode), rel (LinkTypes), rev (LinkTypes), tabindex (Number), type (ContentType)	(PCDATA | Inline -a)

The List Module

The List Module is made up of the elements necessary for creating ordered, unordered, and definition lists. The following table contains the List module:

Elements	Attributes	Minimal Content Model
dl	Common	(dt | dd)+
dt	Common	(PCDATA | Inline)*
dd	Common	(PCDATA | Flow)*
ol	Common	li+
ul	Common	li+
li	Common	(PCDATA | Flow)*

Applet Module (deprecated)

The Applet module has been deprecated in favor of the Object module. The Applet module contains the elements necessary for embedding "external applications" in a document. The following table lists the elements and attributes of the Applet module:

Elements	Attributes	Minimal Content Model
applet	Core alt (Text), archive (CDATA), code (CDATA), codebase (URI), height* (Length), object (CDATA), width* (Length)	(PCDATA | Flow | param)*
param	id (ID), name* (CDATA), type (ContentType), value (CDATA), valuetype ("data"*|"ref"|"object")	Empty

Text Extension Modules

This group of modules contains elements and attributes that also are used with text. The modules in this group include the Presentation module, the Edit module, and the Bidirectional module.

Presentation Module

The elements and attributes of the Presentation module affect the visual appearance of the text—for example: bold, italicized, and underlined text elements. The following table lists the elements and attributes of the presentation module:

Elements	Attributes	Minimal Content Model
b	Common	(PCDATA | Inline)*
big	Common	(PCDATA | Inline)*
hr	Common	Empty
i	Common	(PCDATA | Inline)*
small	Common	(PCDATA | Inline)*
sub	Common	(PCDATA | Inline)*
sup	Common	(PCDATA | Inline)*
tt	Common	(PCDATA | Inline)*

The Edit Module

The elements and attributes that make up the Edit module are used when editing a document. These elements enable an author to indicate deleted and inserted text. The following table lists the Edit module's elements and attributes:

Elements	Attributes	Minimal Content Model
Del	Common, cite (URI), datetime (Datetime)	(PCDATA | Flow)*
Ins	Common, cite (URI), datetime (Datetime)	(PCDATA | Flow)*

The Bidirectional Text Module

The Bidirectional Text module contains an element that can be used to specify the direction of text (Example: "ltr" = left-to-right). The Bi-Di Override, <bdo>, element is the single element making up this module.

Element	Attributes	Minimal Content Model
Bdo	Core, dir ("ltr" | "rtl")	(PCDATA | Inline)*

Forms Modules

Forms modules contain the elements and attributes necessary for creating XHTML forms. There are two modules in this group: the Basic Forms Module, which is a scaled-down version of the second, the Forms module.

Basic Forms Module

The Basic Forms module includes a limited number of form elements. This is for use with XHTML Basic in creating content for alternative technologies (wireless, PDAs, and so on). The following table contains the elements of the Basic Forms module:

Elements	Attributes	Minimal Content Model
`Form`	Common, action (URI), method ("get" | "post")enctype (ContentType)	(Heading | List | Block -form)+
`Input`	Common, accesskey (Character), checked ("checked"), maxlength (Number), name (CDATA), size (Number), src (URI), tabindex (Number), type ("text"*|"password"| "checkbox"|"radio"|"submit"|"reset"|"hidden"), value (CDATA)	Empty
`Label`	Common, accesskey (Character), for (IDREF)	(PCDATA | Inline - label)*
`Select`	Common, multiple ("multiple"), name, size, tabindex	option+
`Option`	Common,selected ("selected"), value (CDATA)	PCDATA
`Textarea`	Common, accesskey (Character), cols* (Number), name (CDATA), rows* (Number), tabindex (Number)	PCDATA

Forms Module

The Forms module contains all the elements and attributes that were part of the HTML 4.0 specification. Listed in the following table are the elements that comprise the Forms module:

Elements	Attributes	Minimal Content Model
`form`	Common, accept (ContentTypes), accept-charset (Charsets), action (URI), method("get"|"post"), enctype (ContentType)	(Heading|List|Block-form|fieldset)+
`input`	Common, accept (ContentTypes), accesskey (Character), alt (Text), checked ("checked"), disabled ("disabled"), maxlength (Number), name (CDATA), readonly ("readonly"), size (Number), src (URI), tabindex (Number), type ("text|"password"|"checkbox"|"button"|"radio"|"submit"|"reset"|"file"|"hidden"|"image"), value (CDATA)	Empty
`select`	Common, disabled("disabled"), multiple("multiple"), name (CDATA), size (Number), tabindex (Number)	(optgroup|option)+
`option`	Common, disabled("disabled"), label (Text), selected("selected"), value (CDATA)	PCDATA

Elements	Attributes	Minimal Content Model
`textarea`	Common, accesskey (Character), cols* (Number), disabled("disabled"), name (CDATA), readonly("readonly"), rows*(Number), tabindex (Number)	PCDATA
`button`	Common, accesskey (Character), disabled("disabled"), name (CDATA), tabindex (Number), type("button"|"submit"|"reset"), value (CDATA)	(PCDATA | Heading | List| Block-Form | Inline - Formctrl)*
`fieldset`	Common	(PCDATA | legend | Flow)
`label`	Common, accesskey (Character), for (IDREF)	(PCDATA | Inline-label)*
`legend`	Common, accesskey (Character)	(PCDATA | Inline)
`optgroup`	Common, disabled("disabled"), label* (Text)	option+

Tables Modules

The Tables module group contains the Basic Tables module and the Tables module. The Basic module is a scaled-down version of the full Table module.

Basic Tables Module

The Basic Tables module includes a limited number of table-related elements. The Basic Tables module is presented in the following table:

Elements	Attributes	Minimal Content Model
`Caption`	Common	(PCDATA | Inline)*
`Table`	Common, summary (Text), width (Length)	caption? tr
`Td`	Common, abbr (Text), align("left"|"center"|"right"), axis (CDATA), colspan (Number), headers (IDREFS), rowspan(Number), scope("row"|"col"), valign("top"|"middle"|"bottom")	(PCDATA | Flow-table)*
`Th`	Common, abbr (Text), align("left"|"center"|"right"), axis (CDATA), colspan (Number), headers (IDREFS), rowspan (Number), scope("row"|"col"), valign("top"|"middle"|"bottom")	(PCDATA | Flow-table)*
`Tr`	Common, align("left"|"center"|"right"), valign("top"|"middle"|"bottom")	(td | th)

Tables Module

The elements of the Tables module include markup that makes tables more accessible to nonvisual browsers. The following table lists the elements and attributes that are part of the Tables module:

Elements	Attributes	Minimal Content Model
`caption`	Common	(PCDATA | Inline)*
`table`	Common, border (Pixels), cellpadding (Length), cellspacing (Length), datapagesize (CDATA), frame("void"|"above"|"below"|"hsides"|"lhs"|"rhs"| "vsides"|"box"|"border"), rules("none"|"groups"| "rows"|"cols"|"all"), summary (Text), width (Length)	caption?,(col* | colgroup*), ((thead?, tfoot?, tbody+)|(tr+))
`td`	Common, abbr (Text), align("left"|"center"|"right"|"justify"|"char"), axis (CDATA), char (Character), charoff (Length), colspan (Number), headers (IDREFS), rowspan (Number), scope("row"|"col"|"rowgroup"|"colgroup"), valign("top"|"middle"|"bottom"|"baseline")	(PCDATA | Flow)*
`th`	Common, abbr (Text), align("left"|"center"| "right"|"justify"|"char"), axis (CDATA), char (Character), charoff (Length), colspan (Number), headers (IDREFS), rowspan (Number), scope ("row"|"col"|"rowgroup", valign("top"|"middle"| "bottom"|"baseline")	(PCDATA | Flow)*
`tr`	Common, align("left"|"center"|"right"|"justify"| "char"), char (Character), charoff (Length), valign("top"|"middle"|"bottom"|"baseline")	(td | th)+
`col`	Common, align("left"|"center"|"right"|"justify"|"char"), char (Character), charoff (Length), span (Number), valign("top"|"middle"|"bottom"|"baseline"), width (MultiLength)	Empty
`colgroup`	Common, align("left"|"center"|"right"|"justify"|"char"), char (Character), charoff (Length), span (Number), valign("top"|"middle"|"bottom"|"baseline"), width (MultiLength)	col*
`tbody`	Common, align("left"|"center"|"right"|"justify"| "char"), char (Character), charoff (Length), valign("top"|"middle"|"bottom"|"baseline")	tr+
`thead`	Common, align("left"|"center"|"right"|"justify"| "char"), char (Character), charoff (Length), valign("top"|"middle"|"bottom"|"baseline")	tr+
`tfoot`	Common, align("left"|"center"|"right"|"justify"| "char"), char (Character), charoff (Length), valign("top"|"middle"|"bottom"|"baseline")	tr+

The Image Module

The Image module enables the document author to embed images and can also be used with client-side image maps. The elements and attributes comprising the Image module are listed in the following table:

Elements	Attributes	Minimal Content Model
`img`	Common, alt (Text), height (Length), longdesc (URI), src (URI), width (Length)	Empty

Client-Side Image Map Module

The Client-Side Image Map module is made up of elements necessary for creating client-side image maps. This module must work in conjunction with either the Image module or another module that supports the <img> element. The elements of the Client-Side Image Map module are included in the following table:

Elements	Attributes	Minimal Content Model
`a&`	coords (CDATA), shape("rect"|"circle"|"poly"|"default")	n/a
`area`	Common, accesskey (Character), alt* (Text), coords (CDATA), href (URI), nohref("nohref"), shape("rect"|"circle"|"poly"|"default"), tabindex (Number)	Empty
`img&`	usemap (IDREF)	n/a
`input&`	usemap(IDREF)	n/a
`map`	I18N, Events, class (NMTOKEN), id* (ID), title (CDATA)	((Heading | Block)| area)
`object&`	usemap (IDREF)	Only when the Object module is included.

Server-Side Image Map Module

The Server-Side Image Map module is necessary if you prefer to keep your image map coordinates on the server rather than on the client's machine. This module also requires that the Image module or another module that supports the <img> element be used in conjunction with it. The following table lists the element of the Server-Side Image Map module:

Elements	Attributes	Minimal Content Model
`img&`	ismap("ismap")	n/a
`input&`	ismap("ismap")	n/a

The Object Module

The Object module can be used to insert "objects" such as Java Applets, audio, video, and even images in a page. In fact, when XHTML 2.0 is fully implemented, the <img /> element will be removed in favor of <object>. This module also will replace the <applet> and <embed> elements.

Elements	Attributes	Minimal Content Model
`object`	Common, archive (URI), classid (URI), codebase (URI), codetype (ContentType), data (URI), declare("declare"), height (Length), name (CDATA), standby (Text), tabindex (Number), type (ContentType), width (Length)	(PCDATA | Flow | param)*
`param`	id (ID), name* (CDATA), type (ContentType), value (CDATA), valuetype("data"|"ref"|"object")	Empty

The Frames Module

This module includes the elements necessary for creating framesets and frames pages. The following table lists the elements of the Frames module:

Elements	Attributes	Minimal Content Model
`frameset`	Core, cols (MultiLength), rows (MultiLength)	(frameset|frame), noframes?
`frame`	Core, frameborder("1"|"0"), longdesc (URI), marginheight (Pixels), marginwidth (Pixels), noresize("noresize"), scrolling("yes"|"no"|"auto"), src (URI)	Empty
`noframes`	Common	body

The Target Module

The Target module works alongside the Frames module. Because it is possible with frames to specify a destination target, this module provides the elements necessary to accomplish this. The following table lists the elements of the Target module:

Elements	Attributes
`a&`	target (CDATA)
`area&`	target (CDATA)
`base&`	target (CDATA)
`link&`	target (CDATA)
`form&`	target (CDATA)

The Iframe Module

The Iframe module is used for creating inline frames. Only a single element, <iframe >, is included in this module. The table that follows includes the element and attributes of the Iframe module:

Elements	Attributes	Minimal Content Model
`Iframe`	Core, frameborder("1"|"0"), height (Length), longdesc (URI), marginheight (Pixels), marginwidth (Pixels), scrolling("yes"|"no"|"auto"), src (URI), width (Length)	(PCDATA|Flow)

The Intrinsic Events Module

The Intrinsic Events module includes attributes that enable certain things to happen when a user performs a particular action. The attributes of this module are listed in the following table:

Elements	Attributes
`a&`	onblur (Script), onfocus (Script)
`area&`	onblur(Script), onfocus (Script)
`frameset&`	onload (Script), onunload (Script)
`form&`	onreset (Script), onsubmit (Script)
`body&`	onload (Script), onunload (Script)
`label&`	onblur (Script), onfocus (Script)
`input&`	onblur (Script), onchange (Script), onfocus (Script), onselect (Script)
`select&`	onblur (Script), onchange (Script), onfocus
`textarea&`	onblur (Script), onchange (Script), onfocus (Script), onselect (Script)
`button&`	onblur (Script), onfocus (Script)

The Metainformation Module

The Metainformation module includes the <meta /> element, enabling the document author to provide descriptive information about the document. The following table lists the element and attributes of the Metainformation module:

Elements	Attributes	Minimal Content Model
`meta`	I18N, content, http-equiv, name, scheme	Empty

The Scripting Module

The Scripting module includes elements necessary for including scripts on a page and also for providing alternate content for those browsers that do not support scripts. Included in the following table are the elements and attributes of the Scripting module:

Elements	Attributes	Minimal Content Model
`noscript`	Common	(Heading | List | Block)
`script`	charset (Charset), defer("defer"), src (URI), type (ContentType), xml:space="preserve"	PCDATA

The Style Sheet Module

The Style Sheet module is for embedding style sheets in a document. This module includes only the <style> element.

Elements	Attributes	Minimal Content Model
`Style`	I18N, media (MediaDesc), title (Text), type (ContentType), xml:space="preserve"	PCDATA

The Link Module

The Link module is used to link a document to "external resources." For example, the <link> element can be used to link to an external style sheet or script. This module is not used for creating hypertext links. For that, you need the Hypertext module. The following table lists the element and attributes of the Link module:

Elements	Attributes	Minimal Content Model
`link`	Common, charset (Charset), href (URI), hreflang (LanguageCode), media (MediaDesc), rel (LinkTypes), rev (LinkTypes), type (ContentType)	Empty

The Base Module

The Base module is used to specify a URI that can be used as a "base" or reference point for relative URIs in a document. This module includes only the <base /> element. The following table lists the element and attribute of the Base module:

Elements	Attributes	Minimal Content Model
`base`	href*(URI)	Empty

The Name Identification Module (Deprecated)

Although the *name* attribute has been deprecated in favor of *id*, there may be occasions when a document needs to include both. The Name Identification module has been created to address that need. The following table lists the elements and attributes of the Name Identification module:

Elements	Attributes	Notes
a&	name (CDATA)	
applet&	name (CDATA)	When the Applet module is selected
form&	name (CDATA)	When the Forms or Basic Forms module is selected
frame&	name (CDATA)	When the Frames module is selected
iframe&	name (CDATA)	When the Iframe module is selected
img&	name (CDATA)	When the Image module is selected
map&	name (CDATA)	When the Client-side Image Map module is selected.

The Legacy Module

The Legacy module might also be called the "History module." It includes the elements and attributes that were deprecated in HTML 4.01 and XHTML 1.0. These modules exist for informational purposes. The elements and attributes contained in them should not be used.

Legacy Module Elements

The following table lists the elements that were deprecated in HTML 4.01 and XHTML 1.0. These elements have been dropped altogether in XHTML 1.1.

Elements	Attributes	Minimal Content Model
basefont	color (Color), face (CDATA), id (ID), size (CDATA)	Empty
center	Common	(PCDATA|Flow)*
dir	Common, compact("compact")	(li)+
font	Core, I18N, color (Color), face (CDATA), size (CDATA)	(PCDATA|Inline)*
isindex	Core, I18N, prompt (Text)	Empty
menu	Common, compact("compact")	(li)+
s	Common	(PCDATA|Inline)*
strike	Common	(PCDATA|Inline)*
u	Common	(PCDATA|Inline)*

Legacy Module Attributes

The following table lists attributes that have either been deprecated altogether, or have been deprecated for use with the listed elements. As with the Legacy module elements, document author should not use these attributes. They have been dropped from XHTML 1.1.

Elements	Attributes	Notes
body&	alink (Color), background (URI), bgcolor (Color), link (Color), text (Color), vlink (Color)	
br&	clear ("left" | "all" | "right" | "none")	
caption&	align ("top" | "bottom" | "left" | "right")	
div&	align ("top" | "bottom" | "left" | "right")	
dl&	compact ("compact"), type (CDATA)	
h1-h6&	align ("left" | "center" | "right" | "justify")	
hr&	align ("left" | "center" | "right" | "justify"), noshade ("noshade"), size (Pixels), width (Pixels)	
img&	align ("left" | "center" | "right" | "justify"), border (Pixels), hspace (Pixels), vspace (Pixels)	
input&	align ("top" | "middle" | "bottom" | "left" | "right")	When the Basic Forms or Forms module is selected
legend&	align ("left" | "center" | "right" | "justify")	When the Forms module is selected
li&	type (CDATA), value (Number)	
ol&	compact("compact"), start (Number), type (CDATA)	
p&	align(left"|"center"|"right"|"justify")	
pre&	width(Number)	
script&	language(CDATA)	When the Scripting module is selected
table&	align("left"|"center"|"right"), bgcolor(Color)	When the Tables module is selected
tr&	bgcolor(Color)	When the Tables module is selected
th&	bgcolor(Color), height(Pixels), nowrap("nowrap"), width(Length)	When the Tables module is selected
td	bgcolor(Color), height(Length) nowrap("nowrap"), width(Pixels)	When the Tables module is selected
ul&	compact("compact"), type(CDATA)	