UNIT - 1
Fundamentals of Web, XHTML – 1

1.1 Internet

1.2 WWW

1.3 Web Browsers

1.4 Web Servers

1.5 URLs

1.6 MIME

1.7 HTTP

1.8 Security

1.9 The Web Programmers Toolbox

1.10 XHTML: Origins and evolution of HTML and XHTML

1.11 Basic syntax

1.12 Standard XHTML document structure

1.13 Basic text markup

1.1 Internet

The **Internet** is a global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a network of networks that consists of millions of private, public, academic, business, and government networks of local to global scope that are linked by a broad array of electronic and optical networking technologies. The Internet carries a vast array of information resources and services, most notably the inter-linked hypertext documents of the World Wide Web (WWW) and the infrastructure to support electronic mail.

Most traditional communications media, such as telephone and television services, are reshaped or redefined using the technologies of the Internet, giving rise to services such as Voice over Internet Protocol (VoIP) and IPTV. Newspaper publishing has been reshaped into Web sites, blogging, and web feeds. The Internet has enabled or accelerated the creation of new forms of human interactions through instant messaging, Internet forums, and social networking sites.

The origins of the Internet reach back to the 1960s when the United States funded research projects of its military agencies to build robust, fault-tolerant and distributed computer networks. This research and a period of civilian funding of a new U.S. backbone by the National Science Foundation spawned worldwide participation in the development of new networking technologies and led to the commercialization of an international network in the mid 1990s, and resulted in the following popularization of countless applications in virtually every aspect of modern human life. As of 2009, an estimated quarter of Earth's population uses the services of the Internet.

1.2 WWW

The **World Wide Web**, abbreviated as **WWW** and commonly known as **the Web**, is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and

navigate between them by using hyperlinks. Using concepts from earlier hypertext systems, English engineer and computer scientist Sir Tim Berners-Lee, now the Director of the World Wide Web Consortium, wrote a proposal in March 1989 for what would eventually become the World Wide Web.[1] He was later joined by Belgian computer scientist Robert Cailliau while both were working at CERN in Geneva, Switzerland. In 1990, they proposed using "HyperText [...] to link and access information of various kinds as a web of nodes in which the user can browse at will", and released that web in December.

"The World-Wide Web (W3) was developed to be a pool of human knowledge, which would allow collaborators in remote sites to share their ideas and all aspects of a common project." If two projects are independently created, rather than have a central figure make the changes, the two bodies of information could form into one cohesive piece of work.

1.3 Web Browsers

A **web browser** is a software application for retrieving, presenting, and traversing information resources on the World Wide Web. An information resource is identified by a Uniform Resource Identifier (URI) and may be a web page, image, video, or other piece of content.[1] Hyperlinks present in resources enable users to easily navigate their browsers to related resources.

Although browsers are primarily intended to access the World Wide Web, they can also be used to access information provided by Web servers in private networks or files in file systems. Some browsers can be also used to save information resources to file systems.

Internet Explorer, the most used web browser

1.4 Web Servers

A **web server** is a computer program that delivers (serves) content, such as web pages, using the Hypertext Transfer Protocol (HTTP), over the World Wide Web. The term web server can also refer to the computer or virtual machine running the program. In large commercial deployments, a server computer running a web server can be rack-mounted with other servers to operate a web farm.

The inside and front of a Dell PowerEdge Web server

1.5 URLs

Uniform Resource Locator (**URL**) is a Uniform Resource Identifier (URI) that specifies where an identified resource is available and the mechanism for retrieving it. In popular usage and in many technical documents and verbal discussions it is often incorrectly used as a synonym for URI,[1]. The best-known example of a URL is the "address" of a web page on the World Wide Web, e.g. http://www.example.com.

1.6 MIME

Multipurpose Internet Mail Extensions (**MIME**) is an Internet standard that extends the format of e-mail to support:

- Text in character sets other than ASCII
- Non-text attachments
- Message bodies with multiple parts
- Header information in non-ASCII character sets

MIME's use, however, has grown beyond describing the content of e-mail to describing content type in general, including for the web (see Internet media type).

Virtually all human-written Internet e-mail and a fairly large proportion of automated e-mail is transmitted via SMTP in MIME format. Internet e-mail is so closely associated with the SMTP and MIME standards that it is sometimes called **SMTP/MIME** e-mail.[1]

The content types defined by MIME standards are also of importance outside of e-mail, such as in communication protocols like HTTP for the World Wide Web. HTTP requires that data be transmitted in the context of e-mail-like messages, although the data most often is not actually e-mail.

MIME is specified in six linked RFC memoranda: RFC 2045, RFC 2046, RFC 2047, RFC 4288, RFC 4289 and RFC 2049, which together define the specifications.

1.7 HTTP

The **Hypertext Transfer Protocol** (**HTTP**) is an Application Layer protocol for distributed, collaborative, hypermedia information systems.[1]

HTTP is a request-response standard typical of client-server computing. In HTTP, web browsers or spiders typically act as clients, while an application running on the computer hosting the web site acts as a server. The client, which submits HTTP requests, is also referred to as the user agent. The responding server, which stores or creates resources such as HTML files and images, may be called the origin server. In between the user agent and origin server may be several intermediaries, such as proxies, gateways, and tunnels.

HTTP is not constrained in principle to using TCP/IP, although this is its most popular implementation platform. Indeed HTTP can be "implemented on top of any other protocol on the Internet, or on other networks." HTTP only presumes a reliable transport; any protocol that provides such guarantees can be used.[2]

Resources to be accessed by HTTP are identified using Uniform Resource Identifiers (URIs)—or, more specifically, Uniform Resource Locators (URLs)—using the http or https URI schemes.

Its use for retrieving inter-linked resources, called hypertext documents, led to the establishment of the World Wide Web in 1990 by English physicist Tim Berners-Lee.

The original version of HTTP, designated HTTP/1.0, was revised in HTTP/1.1. One of the characteristics in HTTP/1.0 was that it uses a separate connection to the same server for every document, while HTTP/1.1 can reuse the same connection to download, for

instance, images for the just served page. Hence HTTP/1.1 may be faster as it takes time to set up such connections.

The standards development of HTTP has been coordinated by the World Wide Web Consortium and the Internet Engineering Task Force (IETF), culminating in the publication of a series of Requests for Comments (RFCs), most notably RFC 2616 (June 1999), which defines HTTP/1.1, the version of HTTP in common use.

Support for pre-standard HTTP/1.1 based on the then developing RFC 2068 was rapidly adopted by the major browser developers in early 1996. By March 1996, pre-standard HTTP/1.1 was supported in Netscape 2.0, Netscape Navigator Gold 2.01, Mosaic 2.7, Lynx 2.5, and in Internet Explorer 3.0. End user adoption of the new browsers was rapid. In March 1996, one web hosting company reported that over 40% of browsers in use on the Internet were HTTP 1.1 compliant. That same web hosting company reported that by June 1996, 65% of all browsers accessing their servers were HTTP/1.1 compliant.[3] The HTTP/1.1 standard as defined in RFC 2068 was officially released in January 1997. Improvements and updates to the HTTP/1.1 standard were released under RFC 2616 in June 1999.

1.8 Security

Computer security is a branch of computer technology known as information security as applied to computers and networks. The objective of computer security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. The term computer system security means the collective processes and mechanisms by which sensitive and valuable information and services are protected from publication, tampering or collapse by unauthorized activities or untrustworthy individuals and unplanned events respectively. The strategies and methodologies of computer security often differ from most other computer technologies because of its somewhat elusive objective of preventing unwanted computer behavior instead of enabling wanted computer behavior.

The technologies of computer security are based on logic. As security is not necessarily the primary goal of most computer applications, designing a program with security in mind often imposes restrictions on that program's behavior.

There are 4 approaches to security in computing, sometimes a combination of approaches is valid:

1. Trust all the software to abide by a security policy but the software is not trustworthy (this is computer insecurity).
2. Trust all the software to abide by a security policy and the software is validated as trustworthy (by tedious branch and path analysis for example).
3. Trust no software but enforce a security policy with mechanisms that are not trustworthy (again this is computer insecurity).
4. Trust no software but enforce a security policy with trustworthy hardware mechanisms.

Computers consist of software executing atop hardware, and a "computer system" is, by frank definition, a combination of hardware, software (and, arguably, firmware, should one choose so separately to categorize it) that provides specific functionality, to include either an explicitly expressed or (as is more often the case) implicitly carried along security policy. Indeed, citing the Department of Defense Trusted Computer System Evaluation Criteria (the TCSEC, or Orange Book)—archaic though that may be —the inclusion of specially designed hardware features, to include such approaches as tagged architectures and (to particularly address "stack smashing" attacks of recent notoriety) restriction of executable text to specific memory regions and/or register groups, was a sine qua non of the higher evaluation classes, to wit, B2 and above.)

Many systems have unintentionally resulted in the first possibility. Since approach two is expensive and non-deterministic, its use is very limited. Approaches one and three lead to failure. Because approach number four is often based on hardware mechanisms and avoids abstractions and a multiplicity of degrees of freedom, it is more practical.

Combinations of approaches two and four are often used in a layered architecture with thin layers of two and thick layers of four.

There are various strategies and techniques used to design security systems. However there are few, if any, effective strategies to enhance security after design. One technique enforces the principle of least privilege to great extent, where an entity has only the privileges that are needed for its function. That way even if an attacker gains access to one part of the system, fine-grained security ensures that it is just as difficult for them to access the rest.

1.9 XHTML: Origins and evolution of HTML and XHTML

HTML

What is HTML?

HTML is a language for describing web pages.

- HTML stands for **H**yper **T**ext **M**arkup **L**anguage
- HTML is not a programming language, it is a **markup language**
- A markup language is a set of **markup tags**
- HTML uses **markup tags** to describe web pages

HTML Tags

HTML markup tags are usually called HTML tags

- HTML tags are keywords surrounded by **angle brackets** like <html>
- HTML tags normally **come in pairs** like and
- The first tag in a pair is the **start tag,** the second tag is the **end tag**
- Start and end tags are also called **opening tags** and **closing tags**

Origins

Tim Berners-Lee

In 1980, physicist Tim Berners-Lee, who was a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an Internet-based hypertext system.[2] Berners-Lee specified HTML and wrote the browser and server software in the last part of 1990. In that year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes,[3] from 1990 he lists[4] "some of the many areas in which hypertext is used", and puts an encyclopedia first.

First specifications

The first publicly available description of HTML was a document called HTML Tags, first mentioned on the Internet by Berners-Lee in late 1991.[5][6] It describes 20 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house SGML based documentation format at CERN. Thirteen of these elements still exist in HTML 4.[7]

HTML is a text and image formatting language used by web browsers to dynamically format web pages. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formatting languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely point effects, and also the separation of structure and processing: HTML has been progressively moved in this direction with CSS.

Berners-Lee considered HTML to be an application of SGML, and it was formally defined as such by the Internet Engineering Task Force (IETF) with the mid-1993 publication of the first proposal for an HTML specification: "Hypertext Markup Language (HTML)" Internet-Draft by Berners-Lee and Dan Connolly, which included an SGML Document Type Definition to define the grammar.[8] The draft expired after six months, but was notable for its acknowledgment of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes.[9] Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.[10]

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.[9] Published as Request for Comments 1866, HTML 2.0 included ideas from the HTML and HTML+ drafts.[11] The 2.0 designation was intended to distinguish the new edition from previous drafts.[12]

Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C).[13] However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). The last HTML specification published by the W3C is the HTML 4.01 Recommendation, published in late 1999. Its issues and errors were last acknowledged by errata published in 2001.

Version history of the standard

HTML version timeline

November 24, 1995

HTML 2.0 was published as IETF RFC 1866. Supplemental RFCs added capabilities:

- November 25, 1995: RFC 1867 (form-based file upload)

- May 1996: RFC 1942 (tables)
- August 1996: RFC 1980 (client-side image maps)
- January 1997: RFC 2070 (internationalization)

In June 2000, all of these were declared obsolete/historic by RFC 2854.

January 1997

HTML 3.2[14] was published as a W3C Recommendation. It was the first version developed and standardized exclusively by the W3C, as the IETF had closed its HTML Working Group in September 1996.[15]

HTML 3.2 dropped math formulas entirely, reconciled overlap among various proprietary extensions, and adopted most of Netscape's visual markup tags. Netscape's blink element and Microsoft's marquee element were omitted due to a mutual agreement between the two companies.[13] A markup for mathematical formulas similar to that in HTML wasn't standardized until 14 months later in MathML.

December 1997

HTML 4.0[16] was published as a W3C Recommendation. It offers three variations:

- Strict, in which deprecated elements are forbidden,
- Transitional, in which deprecated elements are allowed,
- Frameset, in which mostly only frame related elements are allowed;

Initially code-named "Cougar",[17] HTML 4.0 adopted many browser-specific element types and attributes, but at the same time sought to phase out Netscape's visual markup features by marking them as deprecated in favor of style sheets. HTML 4 is an SGML application conforming to ISO 8879 - SGML.[18]

April 1998

HTML 4.0[19] was reissued with minor edits without incrementing the version number.

December 1999

HTML 4.01[20] was published as a W3C Recommendation. It offers the same three variations as HTML 4.0, and its last errata were published May 12, 2001.

May 2000

ISO/IEC 15445:2000[21][22] ("ISO HTML", based on HTML 4.01 Strict) was published as an ISO/IEC international standard. In the ISO this standard falls in the domain of the ISO/IEC JTC1/SC34 (ISO/IEC Joint Technical Committee 1, Subcommittee 34 - Document description and processing languages).[21]

As of mid-2008, HTML 4.01 and ISO/IEC 15445:2000 are the most recent versions of HTML. Development of the parallel, XML-based language XHTML occupied the W3C's HTML Working Group through the early and mid-2000s.

HTML draft version timeline

October 1991

HTML Tags,[5] an informal CERN document listing twelve HTML tags, was first mentioned in public.

June 1992

First informal draft of the HTML DTD, with seven[citation needed] subsequent revisions

November 1992

HTML DTD 1.1 (the first with a version number, based on RCS revisions, which start with 1.1 rather than 1.0), an informal draft

June 1993

Hypertext Markup Language was published by the IETF IIIR Working Group as an Internet-Draft (a rough proposal for a standard). It was replaced by a second version [1] one month later, followed by six further drafts published by IETF itself [2] that finally led to HTML 2.0 in RFC1866

November 1993

HTML+ was published by the IETF as an Internet-Draft and was a competing proposal to the Hypertext Markup Language draft. It expired in May 1994.

April 1995 (authored March 1995)

HTML 3.0 was proposed as a standard to the IETF, but the proposal expired five months later without further action. It included many of the capabilities that were in Raggett's HTML+ proposal, such as support for tables, text flow around figures, and the display of complex mathematical formulas.[25]

W3C began development of its own Arena browser for testing support for HTML 3 and Cascading Style Sheets, but HTML 3.0 did not succeed for several reasons. The draft was considered very large at 150 pages and the pace of browser development, as well as the number of interested parties, had outstripped the resources of the IETF.[13] Browser vendors, including Microsoft and Netscape at the time, chose to implement different subsets of HTML 3's draft features as well as to introduce their own extensions to it.[13] (See Browser wars) These included extensions to control stylistic aspects of documents, contrary to the "belief [of the academic engineering community] that such things as text color, background texture, font size and font face were definitely outside the scope of a language when their only intent was to specify how a document would be organized." Dave Raggett, who has been a W3C Fellow for many years has commented for example, "To a certain extent, Microsoft built its business on the Web by extending HTML features."

January 2008

HTML 5 was published as a Working Draft by the W3C.

Although its syntax closely resembles that of SGML, HTML 5 has abandoned any attempt to be an SGML application, and has explicitly defined its own "html" serialization, in addition to an alternative XML-based XHTML 5 serialization.[27]

XHTML versions

Main article: XHTML

XHTML is a separate language that began as a reformulation of HTML 4.01 using XML 1.0. It continues to be developed:

- XHTML 1.0, published January 26, 2000 as a W3C Recommendation, later revised and republished August 1, 2002. It offers the same three variations as HTML 4.0 and 4.01, reformulated in XML, with minor restrictions.
- XHTML 1.1, published May 31, 2001 as a W3C Recommendation. It is based on XHTML 1.0 Strict, but includes minor changes, can be customized, and is reformulated using modules from Modularization of XHTML, which was published April 10, 2001 as a W3C Recommendation.
- XHTML 2.0,. There is no XHTML 2.0 standard. XHTML 2.0 is incompatible with XHTML 1.x and, therefore, would be more accurate to characterize as an XHTML-inspired new language than an update to XHTML 1.x.
- XHTML 5, which is an update to XHTML 1.x, is being defined alongside HTML 5 in the HTML 5 draft.

XHTML

XHTML (Extensible Hypertext Markup Language) is a family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML), the language in which web pages are written.

While HTML (prior to HTML5) was defined as an application of Standard Generalized Markup Language (SGML), a very flexible markup language framework, XHTML is an application of XML, a more restrictive subset of SGML. Because XHTML documents need to be well-formed, they can be parsed using standard XML parsers—unlike HTML, which requires a lenient HTML-specific parser.

XHTML 1.0 became a World Wide Web Consortium (W3C) Recommendation on January 26, 2000. XHTML 1.1 became a W3C Recommendation on May 31, 2001. XHTML5 is undergoing development as of September 2009, as part of the HTML5 specification.

Origin

XHTML 1.0 is "a reformulation of the three HTML 4 document types as applications of XML 1.0". The World Wide Web Consortium (W3C) also continues to maintain the HTML 4.01 Recommendation, and the specifications for HTML5 and XHTML5 are being actively developed. In the current XHTML 1.0 Recommendation document, as published and revised to August 2002, the W3C commented that, "The XHTML family is the next step in the evolution of the Internet. By migrating to XHTML today, content developers can enter the XML world with all of its attendant benefits, while still remaining confident in their content's backward and future compatibility."[1]

However, in 2004, the Web Hypertext Application Technology Working Group (WHATWG) formed, independently of the W3C, to work on advancing ordinary HTML not based on XHTML. Most major browser vendors were unwilling to implement the features in new W3C XHTML 2.0 drafts, and felt that they didn't serve the needs of modern web development. The WHATWG eventually began working on a standard that supported both XML and non-XML serializations, HTML 5, in parallel to W3C standards such as XHTML 2. In 2007, the W3C's HTML working group voted to officially recognize HTML 5 and work on it as the next-generated HTML standard. In 2009, the W3C allowed the XHTML 2 Working Group's charter to expire, acknowledging that HTML 5 would be the sole next-generation HTML standard, including both XML and non-XML serializations.

Evolution

XHTML 1.0

December 1998 saw the publication of a W3C Working Draft entitled Reformulating HTML in XML. This introduced Voyager, the codename for a new markup language based on HTML 4 but adhering to the stricter syntax rules of XML. By February 1999 the specification had changed name to XHTML 1.0: The Extensible HyperText Markup Language, and in January 2000 it was officially adopted as a W3C Recommendation.[29]

There are three formal DTDs for XHTML 1.0, corresponding to the three different versions of HTML 4.01:

- **XHTML 1.0 Strict** is the XML equivalent to strict HTML 4.01, and includes elements and attributes that have not been marked deprecated in the HTML 4.01 specification.
- **XHTML 1.0 Transitional** is the XML equivalent of HTML 4.01 Transitional, and includes the presentational elements (such as center, font and strike) excluded from the strict version.
- **XHTML 1.0 Frameset** is the XML equivalent of HTML 4.01 Frameset, and allows for the definition of frameset documents—a common Web feature in the late 1990s.

The second edition of XHTML 1.0 became a W3C Recommendation in August 2002.[30]

Modularization of XHTML

Modularization provides an abstract collection of components through which XHTML can be subsetted and extended. The feature is intended to help XHTML extend its reach onto emerging platforms, such as mobile devices and Web-enabled televisions. The initial draft of Modularization of XHTML became available in April 1999, and reached Recommendation status in April 2001.

The first XHTML Family Markup Languages to be developed with this technique were XHTML 1.1 and XHTML Basic 1.0. Another example is XHTML-Print (W3C Recommendation, September 2006), a language designed for printing from mobile devices to low-cost printers.

In October 2008 Modularization of XHTML was superseded by XHTML Modularization 1.1, which adds an XML Schema implementation.

XHTML 1.1—Module-based XHTML

XHTML 1.1 evolved out of the work surrounding the initial Modularization of XHTML specification. The W3C released a first draft in September 1999; Recommendation status was reached in May 2001. The modules combined within XHTML 1.1 effectively recreate XHTML 1.0 Strict, with the addition of ruby annotation elements (ruby, rbc, rtc, rb, rt and rp) to better support East-Asian languages. Other changes include removal of the lang attribute (in favour of xml:lang), and removal of the name attribute from the a and map elements.

Although XHTML 1.1 is largely compatible with XHTML 1.0 and HTML 4, in August 2002 the HTML WG (renamed to XHTML2 WG since) issued a Working Group Note advising that it should not be transmitted with the HTML media type. With limited browser support for the alternate application/xhtml+xml media type, XHTML 1.1 proved unable to gain widespread use. In January 2009 a second edition of the document (XHTML Media Types - Second Edition, not to be confused with the XHTML 1.1 - 2nd ed) was issued, relaxing this restriction and allowing XHTML 1.1 to be served as text/html.

XHTML 1.1 Second Edition (W3C Proposed Edited Recommendation) was issued on 7 May 2009 and rescinded on 19 May 2009. (This does not affect the text/html media type usage for XHTML 1.1 as specified in the: XHTML Media Types - Second Edition)

XHTML Basic and XHTML-MP

To support constrained devices, XHTML Basic was created by the W3C; it reached Recommendation status in December 2000. XHTML Basic 1.0 is the most restrictive version of XHTML, providing a minimal set of features that even the most limited devices can be expected to support.

The Open Mobile Alliance and its predecessor the WAP Forum released three specifications between 2001 and 2006 that extended XHTML Basic 1.0. Known as XHTML Mobile Profile or XHTML-MP, they were strongly focused on uniting the

differing markup languages used on mobile handsets at the time. All provide richer form controls than XHTML Basic 1.0, along with varying levels of scripting support.

XHTML Basic 1.1 became a W3C Recommendation in July 2008, superseding XHTML-MP 1.2. XHTML Basic 1.1 is almost but not quite a subset of regular XHTML 1.1. The most notable addition over XHTML 1.1 is the inputmode attribute—also found in XHTML-MP 1.2—which provides hints to help browsers improve form entry.

XHTML 1.2

The XHTML 2 Working Group is considering the creation of a new language based on XHTML 1.1. If XHTML 1.2 is created, it will include WAI-ARIA and role attributes to better support accessible web applications, and improved Semantic Web support through RDFa. The inputmode attribute from XHTML Basic 1.1, along with the target attribute (for specifying frame targets) may also be present. It's important to note that the XHTML2 WG have not yet been chartered to carry out the development of XHTML1.2 and the W3C has announced that it does not intend to recharter the XHTML2 WG, this means that the XHTML1.2 proposal may not eventuate.

XHTML 2.0

Between August 2002 and July 2006 the W3C released the first eight Working Drafts of XHTML 2.0, a new version of XHTML able to make a clean break from the past by discarding the requirement of backward compatibility. This lack of compatibility with XHTML 1.x and HTML 4 caused some early controversy in the web developer community. Some parts of the language (such as the role and RDFa attributes) were subsequently split out of the specification and worked on as separate modules, partially to help make the transition from XHTML 1.x to XHTML 2.0 smoother. A ninth draft of XHTML 2.0 was expected to appear in 2009, but on July 2, 2009, the W3C decided to let the XHTML2 Working Group charter expire by that year's end, effectively halting any further development of the draft into a standard.

New features introduced by XHTML 2.0 include:

- HTML forms will be replaced by XForms, an XML-based user input specification allowing forms to be displayed appropriately for different rendering devices.
- HTML frames will be replaced by XFrames.
- The DOM Events will be replaced by XML Events, which uses the XML Document Object Model.
- A new list element type, the nl element type, will be included to specifically designate a list as a navigation list. This will be useful in creating nested menus, which are currently created by a wide variety of means like nested unordered lists or nested definition lists.
- Any element will be able to act as a hyperlink, e. g., <li href="articles.html">Articles, similar to XLink. However, XLink itself is not compatible with XHTML due to design differences.
- Any element will be able to reference alternative media with the src attribute, e. g., London Bridge is the same as <object src="lbridge.jpg" type="image/jpeg"> London Bridge </object>.
- The alt attribute of the img element has been removed: alternative text will be given in the content of the img element, much like the object element, e. g., <img src="hms_audacious.jpg">HMS Audacious</img>.
- A single heading element (h) will be added. The level of these headings are determined by the depth of the nesting. This allows the use of headings to be infinite, rather than limiting use to six levels deep.
- The remaining presentational elements i, b and tt, still allowed in XHTML 1.x (even Strict), will be absent from XHTML 2.0. The only somewhat presentational elements remaining will be sup and sub for superscript and subscript respectively, because they have significant non-presentational uses and are required by certain languages. All other tags are meant to be semantic instead (e. g. for **strong or bolded** text) while allowing the user agent to control the presentation of elements via CSS.
- The addition of RDF triple with the property and about attributes to facilitate the conversion from XHTML to RDF/XML.

HTML5—Vocabulary and APIs for HTML5 and XHTML5

Main article: HTML5

HTML5 initially grew independently of the W3C, through a loose group of browser manufacturers and other interested parties calling themselves the WHATWG, or Web Hypertext Application Technology Working Group. The WHATWG announced the existence of an open mailing list in June 2004, along with a website bearing the strapline ―Maintaining and evolving HTML since 2004.‖ The key motive of the group was to create a platform for dynamic web applications; they considered XHTML 2.0 to be too document-centric, and not suitable for the creation of internet forum sites or online shops.

In April 2007, the Mozilla Foundation and Opera Software joined Apple in requesting that the newly rechartered HTML Working Group of the W3C adopt the work, under the name of HTML 5. The group resolved to do this the following month, and the First Public Working Draft of HTML 5 was issued by the W3C in January 2008. The most recent W3C Working Draft was published in June 2008.

HTML5 has both a regular text/html serialization and an XML serialization, which is known as XHTML5. In addition to the markup language, the specification includes a number of application programming interfaces. The Document Object Model is extended with APIs for editing, drag-and-drop, data storage and network communication.

The language is more compatible with HTML 4 and XHTML 1.x than XHTML 2.0, due to the decision to keep the existing HTML form elements and events model. It adds many new elements not found in XHTML 1.x, however, such as section and aside. (The XHTML 1.2 equivalent (which (X)HTML5 replaces) of these structural elements would be and .)

As of 2009-09-03, the latest editor‘s draft includes WAI-ARIA support.

1.11 Basic syntax

Writing valid HTML (or XHTML) is not a terribly difficult task once you know what the rules are, although the rules are slightly more stringent in XHTML than in HTML. The list below provides a quick reference to the rules that will ensure your markup is well-formed and valid. Note that there are other differences between HTML and XHTML which go beyond simple syntax requirements; those differences are covered in HTML Versus XHTML.

The Document Tree

A web page is, at its heart, little more than a collection of HTML elements—the defining structures that signify a paragraph, a table, a table cell, a quote, and so on. The element is created by writing an opening tag, and completed by writing a closing tag. In the case of a paragraph, you'd create a p element by typing Content goes here .

The elements in a web page are contained in a tree structure in which html is the root element that splits into the head and body elements (as explained in Basic Structure of a Web Page). An element may contain other nested elements (although this very much depends on what the parent element is; for example, a p element can contain span, em, or strong elements, among others). Where this occurs, the opening and closing tags must be symmetrical. If an opening paragraph tag is followed by the opening em element, the closing tags must appear in the reverse order, like so: Content goes here, and some of it needs emphasis too . If you were to type Content goes here, and some of it needs emphasis too , you'd have created invalid markup.

Case Sensitivity

In HTML, tag names are case insensitive, but in XHTML they're case sensitive. As such, in HTML, you can write the markup in lowercase, mixed case, or uppercase letters. So this is a paragraph , as is <P>this example</P>, and even <P>this markup would be valid . In XHTML, however, you must use lowercase for markup: This is a valid paragraph in XHTML .

Opening and Closing Tags

In HTML, it‘s possible to omit some closing tags (check each element‘s reference to see whether an HTML closing tag is required), so this is valid markup: This is my first paragraph. This is my second paragraph. And here‘s the last one..

In XHTML, all elements must be closed. Hence the paragraph example above would need to be changed to: This is my first paragraph. This is my second paragraph. And here‘s the last one.

As well as letting you omit some closing tags, HTML allows you to omit start tags—but only on the html, head, body, and tbody elements. This is not a recommended practice, but is technically possible.

For empty elements such as img, XHTML (that is not served with the application/xhtml+xml) requires us to use the XML empty element syntax: <elementname attribute="attributevalue"/>

If serving the document as application/xhtml+xml, it‘s also valid to close empty elements using a start and end tag, for example the img element, as <img></img>

Readability Considerations

A browser doesn‘t care whether you use a single space to separate attributes, ten spaces, or even complete line breaks; it doesn‘t matter, as long as some space is present. As such, all of the examples below are perfectly acceptable (although the more spaces you include, the larger your web page‘s file size will be—each occurrence of whitespace takes up additional bytes—so the first example is still the most preferable):

```
<img src="/images/burj.jpg" alt="Burj Al Arab, iconic hotel in
Dubai" class="gallery"/>

<img
    src="/images/burj.jpg"
```

```
    alt="Burj Al Arab, iconic hotel in Dubai"
    class="gallery"
/>

<img

    src="/images/burj.jpg"

    alt="Burj Al Arab, iconic hotel in Dubai"

    class="gallery"/>
```

In XHTML all attribute values must be quoted, so you'll need to write class="gallery" rather than class=gallery. It's valid to omit the quotes from your HTML, though it may make reading the markup more difficult for developers revisiting old markup (although this really depends on the developer—it's a subjective thing). It's simply easier always to add quotes, rather than to have to remember in which scenarios attribute values require quotes in HTML, as the following piece of HTML demonstrates:

```
<a href="http://example.org"> needs to be quoted because it contains a /
<a href=index.html> acceptable without quotes in HTML
```

Another reason why it's a good idea always to quote your attributes, even if you're using HTML 4.01, is that your HTML editor may be able to provide syntax coloring that makes the code even easier to scan through. Without the quotes, the software may not be able to identify the difference between elements, attributes, and attribute values. This fact is illustrated in Figure , which shows a comparison between quoted and unquoted syntax coloring in the Mac text editor TextMate.

Figure. TextMate‘s syntax coloring taking effect to display quoted attributes

```
<title>My lovely web page</title>
</head>

<body id=splash-page>
    <div id=main-content>
        <h1>This is my lovely web page</h1>
<p>It has lots of lovely content. It has son
blockquote:</p>
```

```
<title>My lovely web page</title>
</head>

<body id="splash-page">
    <div id="main-content">
        <h1>This is my lovely web page</h1>
<p>It has lots of lovely content. It has son
blockquote:</p>
```

Commenting Markup

You may add comments in your HTML, perhaps to make it clear where sections start or end, or to provide a note to remind yourself why you approached the creation of a page in a certain way. What you use comments for isn‘t important, but the way that you craft a comment is important. The HTML comment looks like this: <!-- this is a comment -->. It‘s derived from SGML, which starts with an <! and ends with an >; the actual comment is, in effect, inside the opening -- and the closing -- parts. These hyphens tell the browser when to start ignoring text content, and when to start paying attention again. The fact that the double hyphen -- characters signify the beginning and end of the comment means that you should not use double hyphens anywhere inside a comment, even if you believe that your usage of these characters conforms to SGML rules. Single hyphens are allowed, however.

The markup below shows examples of good and bad HTML comments—see the remark associated with each example for more information:

```
<p>Take the next right.<!-- Look out for the
   signpost for 'Castle' --></p> a valid comment
```

Take the next right.<!-- Look out for -- Castle -->
not a valid comment; the double dashes in the middle could be
misinterpreted as the end of the comment

Take the next right.<!-- Look out for -- -- Castle -->
a valid comment; 'Look out for' is one comment, 'Castle' is another

Take the next right.
<!---------------------------------
This is just asking for trouble. Too
many hyphens! -->
a valid comment; don't use hyphens or <> characters to format comment text

<p <!-- class="lively" -->>Wowzers!
It's not possible to comment out attributes inside an HTML element

1.12 Standard XHTML document structure

An XHTML document consists of three main parts:

- DOCTYPE
- Head
- Body

The basic document structure is:

```
<!DOCTYPE ...>

<html  ... >
<head> ... </head>
```

```
<body> ... </body>
</html>
```

The <head> area contains information about the document, such as ownership, copyright, and keywords; and the <body> area contains the content of the document to be displayed.

Listing 1 shows you how this structure might be used in practice:

Listing 1. An XHTML example

```
1.  <?xml version="1.0"?>
2.  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
    Transitional//EN"  "DTD/xhtml1-transitional.dtd">
3.  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
                lang="en">
4.  <head>
    <title>My XHTML Sample Page</title>

    </head>
5.  <body bgcolor="white">
    <center><h1>Welcome to XHTML !</h1></center>
    </body>
6.  </html>
```

Line 1: Since XHTML is HTML expressed in an XML document, it must include the initial XML declaration <?xml version="1.0"?> at the top of the document.

Line 2: XHTML documents must be identified by one of three standard sets of rules. These rules are stored in a separate document called a Document Type Declaration

(DTD), and are utilized to validate the accuracy of the XHTML document structure. The purpose of a DTD is to describe, in precise terms, the language and syntax allowed in XHTML.

Line 3: The second tag in an XHTML document must include the opening <html> tag with the XML namespace identified by the xmlns=http://www.w3.org/1999/xhtml attribute. The XML namespace identifies the range of tags used by the XHTML document. It is used to ensure that names used by one DTD don't conflict with user-defined tags or tags defined in other DTDs.

Line 4: XHTML documents must include a full header area. This area contains the opening <head> tag and the title tags (<title></title>), and is then completed with the closing </head> tag.

Line 5: XHTML documents must include opening and closing <body></body> tags. Within these tags you can place your traditional HTML coding tags. To be XHTML conformant, the coding of these tags must be well-formed.

Line 6: Finally, the XHTML document is completed with the closing </html> tag.

1.13 Basic text markup

A **markup language** is a modern system for annotating a text in a way that is syntactically distinguishable from that text. The idea and terminology evolved from the "marking up" of manuscripts, i.e. the revision instructions by editors, traditionally written with a blue pencil on authors' manuscripts. Examples are typesetting instructions such as those found in troff and LaTeX, and structural markers such as XML tags. Markup is typically omitted from the version of the text which is displayed for end-user consumption. Some markup languages, like HTML have presentation semantics, meaning their specification prescribes how the structured data is to be presented, but other markup languages, like XML, have no predefined semantics.

A well-known example of a markup language in widespread use today is HyperText Markup Language (HTML), one of the document formats of the World Wide Web. HTML is mostly an instance of SGML (though, strictly, it does not comply with all the rules of SGML) and follows many of the markup conventions used in the publishing industry in the communication of printed work between authors, editors, and printers.