



History of internet and Web Key Terminology

Internet Protocol

The Internet Protocol (IP) is the method or protocol by which data is sent from one computer to another on the Internet. Each computer (known as a host) on the Internet has at least one IP address that uniquely identifies it from all other computers on the Internet.

Stands for "Internet Protocol." IP provides a standard set of rules for sending and receiving data over the Internet. It allows devices running on different platforms to communicate with each other as long as they are connected to the Internet.

In order for a Internet-connected host to be recognized by other devices, it must have an IP address.

The Internet Protocol also provides basic instructions for transferring packets between devices.

Client-Server Model

The client-server model describes how a server provides resources and services to one or more clients. Examples of servers include web servers, mail servers, and file servers. Each of these servers provide resources to client devices, such as desktop computers, laptops, tablets, and smart phones.

When a client requests a connection to a server, the server can either accept or reject the connection. If the connection is accepted, the server establishes and maintains a connection with the client over a specific protocol.

DNS

Stands for "Domain Name System." Domain names serve as memorable names for websites and other services on the Internet. However, computers access Internet devices by their IP addresses. DNS translates domain names into IP addresses, allowing you to access an Internet location by its domain name.



Thanks to DNS, you can visit a website by typing in the domain name rather than the IP address. For example, to visit the Tech Terms Computer Dictionary, you can simply type "techterms.com" in the address bar of your web browser rather than the IP address (67.43.14.98).

To understand how DNS works, you can think of it like the contacts app on your smartphone. When you call a friend, you simply select his or her name from a list. The phone does not actually call the person by name, it calls the person's phone number. DNS works the same way by associating a unique IP address with each domain name.

URL

Stands for "Uniform Resource Locator." A URL is the address of a specific webpage or file on the Internet. For example, the URL of the TechTerms website is "http://techterms.com." The address of this page is "http://techterms.com/definition/url" and includes the following elements:

http:// – the URL prefix, which specifies the protocol used to access the location

techterms.com – the server name or IP address of the server

/definition/url – the path to the directory or file

While all website URLs begin with "http," several other prefixes exist. Below is a list of various URL prefixes:

http : a webpage, website directory, or other file available over HTTP

ftp : a file or directory of files available to download from an FTP server

telnet : a Unix-based computer system that supports remote client connections

mailto : an email address (often used to redirect browsers to an email client)

file : a file located on a local storage device (though not technically a URL because it does not refer to an Internet-based location)

You can manually enter a URL by typing it in the address bar of your web browser. For example, you might enter a website URL printed on a business card to visit the company's website. Most URLs, however appear automatically when you click on a link or open a bookmark. If the server name in the URL is not valid, your browser may display a "Server not found" error. If the path in the URL is incorrect, the server may respond with a 404 error.



NOTE: URLs use forward slashes to denote different directories and cannot contain spaces. Therefore, dashes and underscores are often used to separate words within a web address. If your browser produces an error when you visit a specific webpage, you can double-check the URL for typos or other errors. If you find an error, you can manually edit the URL and press Enter to see if it works.

HTTP (Hypertext Transfer Protocol)

HTTP (Hypertext Transfer Protocol) is the set of rules for transferring files, such as text, graphic images, sound, video, and other multimedia files, on the World Wide Web. As soon as a web user opens their web browser, the user is indirectly making use of HTTP. HTTP is an application protocol that runs on top of the TCP/IP suite of protocols (the foundation protocols for the Internet). The latest version of HTTP is HTTP/2, which was published in May 2015. It is an alternative to its predecessor, HTTP 1.1, but does not make it obsolete.

HTTP vs. HTTPS

HTTPS (HTTP over SSL or HTTP Secure) is the use of Secure Sockets Layer (SSL) or Transport Layer Security (TLS) as a sublayer under regular HTTP application layering. HTTPS encrypts and decrypts user HTTP page requests as well as the pages that are returned by the Web server. The use of HTTPS protects against eavesdropping and man-in-the-middle (MitM) attacks. HTTPS was developed by Netscape.

Migrating from HTTP to HTTPS is regarded as good for security.

web server

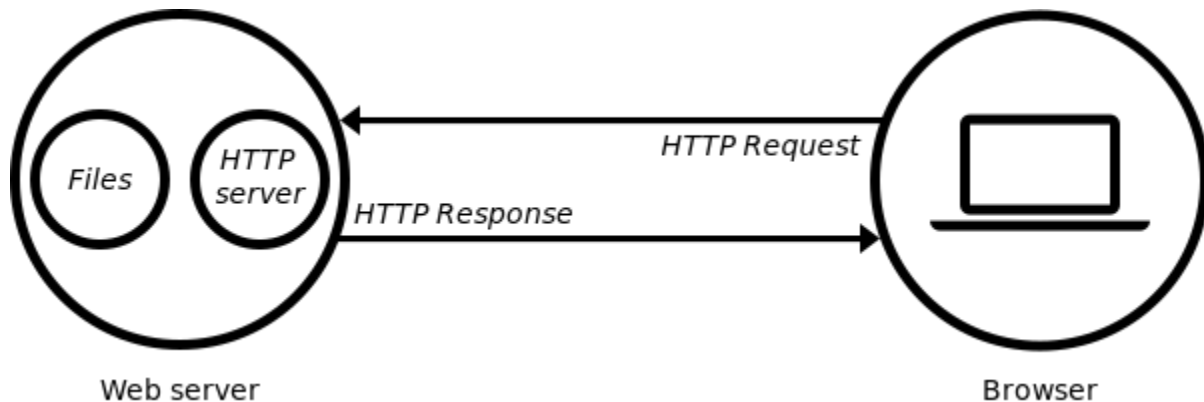
The term web server can refer to hardware or software, or both of them working together.

On the hardware side, a web server is a computer that stores web server software and a website's component files. (for example, HTML documents, images, CSS stylesheets, and JavaScript files) A web server connects to the Internet and supports physical data interchange with other devices connected to the web.

On the software side, a web server includes several parts that control how web users access hosted files. At a minimum, this is an HTTP server. An HTTP server is software that understands URLs (web addresses) and HTTP (the protocol your browser uses to view



WebPages). An HTTP server can be accessed through the domain names of the websites it stores, and it delivers the content of these hosted websites to the end user's device.



To publish a website, you need either a static or a dynamic web server.

A static web server, or stack, consists of a computer (hardware) with an HTTP server (software). We call it "static" because the server sends its hosted files as-is to your browser.

A dynamic web server consists of a static web server plus extra software, most commonly an application server and a database. We call it "dynamic" because the application server updates the hosted files before sending content to your browser via the HTTP server.

Introduction to HTML5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web. HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.

New Features

HTML5 introduces a number of new elements and attributes that can help you in building modern websites. Here is a set of some of the most prominent features introduced in HTML5.



- New Semantic Elements : These are like <header>, <footer>, and <section>.
- Forms 2.0 : Improvements to HTML web forms where new attributes have been introduced for <input> tag.
- Persistent Local Storage : To achieve without resorting to third-party plugins.
- WebSocket : A next-generation bidirectional communication technology for web applications.
- Server-Sent Events : HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).
- Canvas : This supports a two-dimensional drawing surface that you can program with JavaScript.
- Audio & Video : You can embed audio or video on your WebPages without resorting to third-party plugins.
- Geolocation : Now visitors can choose to share their physical location with your web application.
- Microdata : This lets you create your own vocabularies beyond HTML5 and extend your web pages with custom semantics.

New structural elements of HTML5

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title></title>
  </head>
  <body>
    <header> ... </header>
    <nav> ... </nav>
    <aside> ... </aside>
    <article>
      <section> ... </section>
    </article>
    <footer> ... </footer>
  </body>
</html>
```



The DOCTYPE

DOCTYPEs in older versions of HTML were longer because the HTML language was SGML based and therefore required a reference to a DTD. HTML 5 authors would use simple syntax to specify DOCTYPE as follows:

```
<!DOCTYPE html>
```

Character Encoding

HTML 5 authors can use simple syntax to specify Character Encoding as follows –

```
<meta charset="UTF-8">
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML, CSS, JavaScript">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Headers

The header element is used for introductory material that typically appears at the beginning of a web page or at the top of a section or article. There is no specified list of what a header must or should contain; anything that makes sense as the introduction to a page or section is acceptable. In the following example, the document header includes a logo image, the site title, and navigation.

```
<header>
  
  <hgroup>
    <h1>Nuts about Web Fonts</h1>
    <h2>News from the Web Typography Front</h2>
  </hgroup>
  <nav>
    <ul>
      <li><a href="">Home</a></li>
      <li><a href="">Blog</a></li>
      <li><a href="">Shop</a></li>
    </ul>
  </nav>
</header>
```



When used in an individual article, the header might include the article title, author, and the publication date, as shown here:

```
<article>
  <header>
    <h1>More about WOFF</h1>
    <p>by Jennifer Robbins, <time datetime="11-11-2011"
      pubdate>November 11, 2011</time></p>
  </header>
  <p>...article content starts here...</p>
</article>
```

Footers

The footer element is used to indicate the type of information that typically comes at the end of a page or an article, such as its author, copyright information, related documents, or navigation. The footer element may apply to the entire document, or it could be associated with a particular section or article. If the footer is contained directly within the body element, either before or after all the other body content, then it applies to the entire page or application. If it is contained in a sectioning element (section, article, nav, or aside), it is parsed as the footer for just that section. Note that although it is called “footer,” there is no requirement that it come last in the document or sectioning element. It could also appear at or near the beginning if it makes semantic sense. In this simple example we see the typical information listed at the bottom of an article or blog post marked up as a footer.

```
<footer>
  <p><small>Copyright &copy;2012 Jennifer Robbins.</small></p>
  <nav>
    <ul>
      <li><a href="">Previous</a></li>
      <li><a href="">Next</a></li>
    </ul>
  </nav>
</footer>
```



Navigation

The new nav element gives developers a semantic way to identify navigation for a site. Earlier in this chapter, we saw an unordered list that might be used as the top-level navigation for a font catalog site. Wrapping that list in a nav element makes its purpose explicitly clear.

```
<nav>
  <ul>
    <li><a href="">Serif</a></li>
    <li><a href="">Sans-serif</a></li>
    <li><a href="">Script</a></li>
    <li><a href="">Display</a></li>
    <li><a href="">Dingbats</a></li>
  </ul>
</nav>
```

Not all lists of links should be wrapped in nav tags, however. The spec makes it clear that it should be used for links that provide primary navigation around a site or a lengthy section or article. The nav element may be especially helpful from an accessibility perspective. Once screen readers and other devices become HTML5-compatible, users can easily get to or skip navigation sections without a lot of hunting around.

Sections and articles

Long documents are easier to use when they are divided into smaller parts. For example, books are divided into chapters, and newspapers have sections for local news, sports, comics, and so on. To divide long web documents into thematic sections, use the aptly named section element. Sections typically have a heading (inside the section element) and any other content that has a meaningful reason to be grouped together.

The section element has a broad range of uses, from dividing a whole page into major sections or identifying thematic sections within a single article. In the following example, a document with information about typography resources has been divided into two sections based on resource type.



```
<section>
  <h2>Typography Books</h2>
  <ul>
    <li>...</li>
  </ul>
</section>
<section>
  <h2>Online Tutorials</h2>
  <p>These are the best tutorials on the web.</p>
  <ul>
    <li>...</li>
  </ul>
</section>
```

Use the article element for self-contained works that could stand alone or be reused in a different context (such as syndication). It is useful for magazine or newspaper articles, blog posts, comments, or other items that could be extracted for external use. You can think of it as a specialized section element that answers the question “Could this appear on another site and make sense?” with “yes.”

To make things interesting, a long article could be broken into a number of sections, as shown here:

```
<article>
  <h1>Get to Know Helvetica</h1>
  <section>
    <h2>History of Helvetica</h2>
    <p>...</p>
  </section>
  <section>
    <h2>Helvetica Today</h2>
    <p>...</p>
  </section>
</article>
```



Conversely, a section in a web document might be comprised of a number of articles.

```
<section id="essays">
  <article>
    <h1>A Fresh Look at Futura</h1>
    <p>...</p>
  </article>
  <article>
    <h1>Getting Personal with Humanist</h1>
    <p>...</p>
  </article>
</section>
```

Aside (sidebars)

The aside element identifies content that is related but tangential to the surrounding content. In print, its equivalent is a sidebar, but they couldn't call the element sidebar, because putting something on the "side" is a presentational description, not semantic. Nonetheless, a sidebar is a good mental model for using the aside element. aside can be used for pull quotes, background information, lists of links, callouts, or anything else that might be associated with (but not critical to) a document.

In this example, an aside element is used for a list of links related to the main article.

```
<p>We now have a number of methods for using beautiful fonts on web
pages...</p>
<aside>
  <h2>Web Font Resources</h2>
  <ul>
    <li>
      <a href="http://typekit.com/">Typekit</a>
    </li>
    <li>
      <a href="http://www.google.com/webfonts">Google Fonts</a>
    </li>
  </ul>
</aside>
```



Heading groups

HTML5 includes the hgroup element for identifying a stack of headings as a group.* Browsers that support hgroup know to count only the highest-ranked heading in the outline and ignore the rest.

```
<hgroup>
  <h1>Creating a Simple Page</h1>
  <h2>(HTML Overview)</h2>
</hgroup>
```

Figures

The figure element is used for content that illustrates or supports some point in the text. A figure may contain an image, a video, a code snippet, text, or even a table—pretty much anything that can go in the flow of web content—and should be treated and referenced as a self-contained unit. That means if a figure is removed from its original placement in the main flow (to a sidebar or appendix, for example), both the figure and the main flow should continue to make sense.

Although it is possible to simply drop an image into text, wrapping it in figure tags makes its purpose explicitly clear. It also allows you to apply special styles to figures but not to other images on the page.

```
<figure>
  
  <figcaption>
    Sample CSS rule.
  </figcaption>
</figure>
```

A caption can be attached to the figure using the optional figcaption element above or below the figure content.



Addresses

Last, and well, least, is the address element that is used to create an area for contact information for the author or maintainer of the document. It is generally placed at the end of the document or in a section or article within a document. An address would be right at home in a footer element.

It is important to note that the address element should not be used for any old address on a page, such as mailing addresses. It is intended specifically for author contact information (although that could potentially be a mailing address). Following is an example of its intended use. The “a href” parts are the markup for links

```
<address>
  Contributed by
  <a href="../authors/robbins/">Jennifer Robbins</a>,
  <a href="http://www.oreilly.com/">O'Reilly Media</a>
</address>
```

Highlighted text

The new mark element indicates a word that may be considered especially relevant to the reader. One might use it to call out a search term in a page of results, to manually call attention to a passage of text, indicate the current page in a series. Some designers (and browsers) give marked text a light colored background as though it was marked with a highlighter marker.

```
<p> ... PART I. ADMINISTRATION OF THE GOVERNMENT. TITLE IX.
  TAXATION. CHAPTER 65C. MASS.
  <mark>ESTATE TAX</mark>.
  Sect. 2. Computation of
  <mark>estate tax</mark>.
</p>
```