GLOBAL EDITION

Computer Science

An Overview

THIRTEENTH EDITION

J. Glenn Brookshear
Dennis Brylow

Computer Science

AN OVERVIEW

13th Edition

Global Edition

J. Glenn Brookshear
(Author Emeritus)
and
Dennis Brylow
Marquette University

Pearson
330 Hudson Street, NY NY 10013

4.2 Questions & Exercises

1. What is the purpose of tier-1 and tier-2 ISPs? What is the purpose of access ISPs?
2. What is DNS?
3. What bit pattern is represented by 3.6.9 in dotted decimal notation? Express the bit pattern 0001010100011100 using dotted decimal notation.
4. In what way is the structure of a mnemonic address of a computer on the Internet (such as `overthruster.propulsion.yoyodyne.com`) similar to a traditional postal address? Does this same structure occur in IP addresses?
5. Name three types of servers found on the Internet and tell what each does.
6. What aspects of network communication are described by a protocol?
7. In what way do the P2P and multicast approaches to Internet radio broadcast differ from N-unicast?
8. What criteria should one consider when choosing one of the four types of VoIP?

4.3 The World Wide Web

The World Wide Web had its origins in the work of Tim Berners-Lee, who realized the potential of combining internet technology with the concept of linked documents, called **hypertext**. His first software for implementing the Web was released in December 1990. While this early prototype did not yet support multimedia data, it included the key components of what we now recognize as the World Wide Web: a hypertext document format for embedding **hyperlinks** to other documents; a protocol for transferring hypertext across the network, and a server process that supplied hypertext pages upon request. From this humble beginning, the Web quickly grew to support images, audio and video, and by the mid 1990s had become the dominant application powering the growth of the Internet.

Web Implementation

Software packages that allow users to access hypertext on the Internet fall into one of two categories: **browsers** and **webservers**. A browser resides on the user's computer and is charged with the tasks of obtaining materials requested by the user and presenting these materials to the user in an organized manner.

Common Internet browsers include Chrome, Firefox, Safari and Internet Explorer. The webserver resides on a computer containing hypertext documents to be accessed. Its task is to provide access to the documents under its control as requested by clients (browsers). Common webserver software packages include Apache, Microsoft IIS, and Nginx. Hypertext documents are normally transferred between browsers and webservers using a protocol known as the Hypertext Transfer Protocol (HTTP).

In order to locate and retrieve documents on the Web, each document is given a unique address called a **Uniform Resource Locator (URL)**. Each URL contains the information needed by a browser to contact the proper server and request the desired document. Thus, to view a webpage, a person first provides his or her browser with the URL of the desired document and then instructs the browser to retrieve and display the document.

A typical URL is presented in Figure 4.8. It consists of four segments: the protocol to use to communicate with the server controlling access to the document, the mnemonic address of the machine containing the server, the directory path needed for the server to find the directory containing the document, and the name of the document itself. In short, the URL in Figure 4.8 tells a browser to contact the webserver on the host known as `eagle.mu.edu` using the protocol HTTP, and to retrieve the document named `Julius_Caesar.html` found within the subdirectory `Shakespeare` within the directory called `authors`.

Sometimes a URL might not explicitly contain all the segments shown in Figure 4.8. For example, if the server does not need to follow a directory path to reach the document, no directory path will appear in the URL. Moreover, sometimes a URL will consist of only a protocol and the mnemonic address of a computer. In these cases, the webserver at that computer will return a predetermined document, typically called a home page, that usually describes the information available at that website. Such shortened URLs provide a simple means of contacting organizations. For example, the URL `http://www.google.com` will lead to the home page of Google, which contains hyperlinks to the services, products, and documents relating to the company.

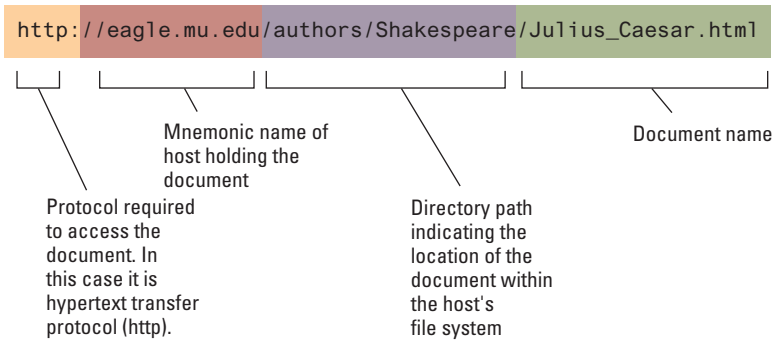


Figure 4.8 A typical URL

To further simplify locating websites, many browsers assume that the HTTP protocol should be used if no protocol is identified. These browsers correctly retrieve the Google home page when given the "URL" consisting merely of `www.google.com`.

HTML

A traditional hypertext document is similar to a text file because its text is encoded character by character using a system such as ASCII or Unicode. The distinction is that a hypertext document also contains special symbols, called **tags**, that describe how the document should appear on a display screen, what multimedia resources (such as images) should accompany the document, and which items within the document are linked to other documents. This system of tags is known as **Hypertext Markup Language (HTML)**.

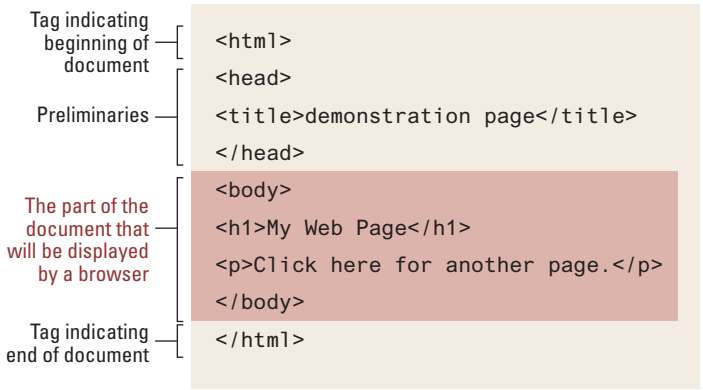
Thus, it is in terms of HTML that an author of a webpage describes the information that a browser needs in order to present the page on the user's screen and to find any related documents referenced by the current page. The process is analogous to adding typesetting directions to a plain typed text (perhaps using a red pen) so that a typesetter will know how the material should appear in its final form. In the case of hypertext, the red markings are replaced by HTML tags, and a browser ultimately plays the role of the typesetter, reading the HTML tags to learn how the text is to be presented on the browser screen.

The HTML-encoded version (called the **source** version) of an extremely simple webpage is shown in Figure 4.9a. Note that the tags are delineated by the symbols < and >. The HTML source document consists of two sections—a head (surrounded by the `<head>` and `</head>` tags) and a body (surrounded by the `<body>` and `</body>` tags). The distinction between the head and body of a webpage is similar to that of the head and body of an interoffice memo. In both cases, the head contains preliminary information about the document (date, subject, etc. in the case of a memo). The body contains the meat of the document, which in the case of a webpage is the material to be presented on the browser screen when the page is displayed.

The World Wide Web Consortium

The World Wide Web Consortium (W3C) was formed in 1994 to promote the World Wide Web by developing protocol standards (known as W3C standards). W3C is headquartered at CERN, the high-energy particle physics laboratory in Geneva, Switzerland. CERN is where the original HTML markup language was developed as well as the HTTP protocol for transferring HTML documents over the Internet. Today, W3C is the source of many standards (including standards for XML and numerous multimedia applications) that lead to compatibility over a wide range of Internet products. You can learn more about W3C via its website at http://www.w3c.org.

a. The page encoded using HTML.



b. The page as it would appear on a computer screen.

Figure 4.9 A simple webpage

The head of the webpage displayed in Figure 4.9a contains only the title of the document (surrounded by "title" tags). This title is only for documentation purposes; it is not part of the page that is to be displayed on the browser screen, although some modern browsers will include the title at the top of the browser window or tab. The material that is displayed on the browser screen is contained in the body of the document.

The first entry in the body of the document in Figure 4.9a is a level-one heading (surrounded by the `<h1>` and `</h1>` tags) containing the text "My Web Page." Being a level-one heading means that the browser should display this text prominently on the screen. The next entry in the body is a paragraph of text (surrounded by the `<p>` and `</p>` tags) containing the text "Click here for another page." Figure 4.9b shows the page as it would be presented by a browser.

In its present form, the page in Figure 4.9 is not fully functional in the sense that nothing will happen when the viewer clicks on the word *here*, even though the page implies that doing so will cause the browser to display another page. To cause the appropriate action, we must link the word *here* to another document.

a. The page encoded using HTML.

```
<html>
<head>
<title>demonstration page</title>
</head>
<body>
<h1>My Web Page</h1>
<p>Click
   <a href="http://crafty.com/demo.html">
   here
   </a>
   for another page.</p>
</body>
</html>
```

Anchor tag containing parameter

Closing anchor tag

b. The page as it would appear on a computer screen.

Figure 4.10 An enhanced simple webpage

Let us suppose that, when the word *here* is clicked, we want the browser to retrieve and display the page at the URL `http://crafty.com/demo.html`. To do so, we must first surround the word *here* in the source version of the page with the tags `<a>` and `</a>`, which are called anchor tags. Inside the opening anchor tag, we insert the parameter

```
href = http://crafty.com/demo.html
```

(as shown in Figure 4.10a), indicating that the hypertext reference (`href`) associated with the tag is the URL following the equal sign (`http://crafty.com/demo.html`). Having added the anchor tags, the webpage will now appear on a browser screen as shown in Figure 4.10b. Note that this is identical to Figure 4.9b except that the word *here* is highlighted by color, indicating that it is a link to another webpage. Clicking on such highlighted terms will cause the

browser to retrieve and display the associated webpage. Thus, it is by means of anchor tags that webpages are linked to each other.

Finally, we should indicate how an image could be included in our simple webpage. For this purpose, let us suppose that a JPEG encoding of the image we want to include is stored as the file named `OurPic.jpg` in the directory `Images` at `Images.com` and is available via the webserver at that location. Under these conditions, we can tell a browser to display the image at the top of the webpage by inserting the image tag `<img src = "http://Images.com/Images/OurPic.jpg">` immediately after the `<body>` tag in the HTML source document. This tells the browser that the image named `OurPic.jpg` should be displayed at the beginning of the document. (The term `src` is short for "source," meaning that the information following the equal sign indicates the source of the image to be displayed.) When the browser finds this tag, it will send a message to the HTTP server at `Images.com` requesting the image called `OurPic.jpg` and then display the image appropriately.

If we moved the image `tag` to the end of the document, just before the `</body>` tag, then the browser would display the image at the bottom of the webpage. There are, of course, more sophisticated techniques for positioning an image on a webpage, but these need not concern us now.

XML

HTML is essentially a notation system by which a text document along with the document's appearance can be encoded as a simple text file. In a similar manner, we can also encode nontextual material as text files—an example being sheet music. At first glance, the pattern of staffs, measure bars, and notes in which music is traditionally represented does not conform to the character-by-character format dictated by text files. However, we can overcome this problem by developing an alternative notation system. More precisely, we could agree to represent the start of a staff by `<staff clef = "treble">`, the end of the staff by `</staff>`, a time signature with the form `<time> 2/4 </time>`, the beginning and ending of a measure by `<measure>` and `</measure>`, respectively, a note such as an eighth note on C as `<notes> eighth C </notes>`, and so on. Then the text

```
<staff clef = "treble"> <key>C minor</key>
<time> 2/4 </time>
<measure> <rest> eighth </rest> <notes> eighth G,
eighth G, eighth G </notes></measure>
<measure> <notes> half E </notes></measure>
</staff>
```

could be used to encode the music shown in Figure 4.11. Using such notation, sheet music could be encoded, modified, stored, and transferred over the Internet as text files. Moreover, software could be written to present the contents of such files in the form of traditional sheet music or even to play the music on a synthesizer.