Chapter 9:
The Internet

Objectives

This laboratory introduces you to the Internet. You should gain an understanding of the following topics:

- A conceptual understanding of the structure and accessibility of information on the World Wide Web (WWW), referred to as the ‘web’, for short.
- Methods (protocols) for serving/accessing data.
- Methods of searching for information on the web (WWW)
- HTML

NB. The Internet is constantly changing as data is updated, removed, or simply moved to a different storage place. Details of this lab will be out of date because of this constant evolution.

Preparation

In order to complete this lab in about 3 hours it is necessary for you to have a clear idea of your objectives. This lab will briefly introduce the idea of a web (WWW) browser, such as Internet Explorer/Firefox/Safari, and then explore how to search for information on the web, and finally provide a glimpse "under the hood" at HTML. It will not be enough for you to simply follow the steps because to properly understand the big picture of the worldwide distributed information services you will need to systematise and classify the navigation and access methods, searching techniques etc. Many details of these features are hidden by the powerful software tools you are using. You should make every effort to understand what you are doing and why you are doing it, rather than just following the steps.

- Read the whole of this lab carefully.
- Attend lectures on this topic
- There are many books about the Internet, any one of which will usefully complement the course materials.

Introduction

There is a lot of hype concerning the Internet. This lab will give you a picture of the reality of these services as they currently exist. There are certainly lots of cool things to do, and no doubt you will be tempted to spend much valuable time exploring this new world of information and entertainment - but a word of warning - please bear in mind the many other academic demands on your time.

Internet Protocols

All applications that work over the Internet rely on the TCP/IP software. TCP stands for Transmission Control Protocol and IP stands for Internet Protocol. A protocol is simply a set of rules for defining how two machines must exchange data. Together the two pieces of software known as TCP/IP define how communications between computers on the Internet - at the level of bits - must take place.

TCP/IP defines what is called a packet switching protocol. Any transmission of data from one location to another is broken up into small "packets" of data. Each packet sent out from the serving computer has the destination address attached to it, along with some other identifying information, in what one might call a wrapper or envelope by analogy with a piece of mail you might post. The packets might (and usually do) take different routes before arriving at the address - you should think of the paths a packet might take as if they were roads in a city with a computer at each intersection choosing which way to send the packet next.

Thus packets might (and usually do) arrive at the receiving computer out of order. The TCP/IP software must use information in the wrapper to put the packets back in order so as to assemble the original collection of data that was sent.

Just as the post office doesn't know what is in the letter you send, neither does the network need to know what data is contained in the packets that are transmitted across the network. Only the information in the wrapper (i.e. on the envelope) is relevant to the TCP/IP software.

Client-Server Architecture and Protocols

Many different applications use the TCP/IP protocols for transmitting and receiving data on a computer network. TCP/IP is the underlying software upon which other protocols, which we shall collectively call client-server protocols, rely.

The client-server approach to communicating between computers is becoming very widespread. Essentially it involves two computer programs - one called the client program which requests data from another computer, and the other called the server program, which responds to the request by "serving" the data.

Generally a computer which predominantly responds to requests from other computers is called a server and a computer which predominantly requests data from other computers is called a client.

Common client programs - i.e. ones which request data - that execute on desktop computers include a web browser and an e-mail program for example. Such client programs must communicate with the server program in a manner defined by the particular protocol involved. We are not referring here to the TCP/IP protocol, which is essentially contained in the wrapper of a packet, but rather we are now examining what is inside the envelope - or rather inside the reassembled packets. Part of this data that is transmitted between computers is the rules by which the client-server programs communicate.

One such client-server protocol is represented by the acronym http, which stands for hypertext transport protocol. This is the most common protocol you will encounter. It is the protocol that was designed to support what we now know as the web.

Other protocols you might encounter are gopher, ftp, telnet, news (nntp) and mail (smtp). A web browser is designed to be compatible with all of these older protocols, as well as the http protocol.

We will briefly examine examples of documents obtained using these protocols, although we will not examine the details of any protocol in depth. The main aim is for you to be an ***aware*** user of the Internet - aware of some of the processes taking place behind the scenes as you use this powerful technology.

The World Wide Web: web browser

Note: This section is optional. It touches on a few aspects of using web browsers and search engines. It will not be tested.

Introduction

Launch Internet Explorer or Firefox.

We assume that you are familiar with at least one web browser, probably Internet Explorer, so we won't bother with the basics of using one.

One feature you might not have made extensive use of is Bookmarking. A Bookmark lets you easily return to a document you encountered while browsing the net. It sure beats hunting around to try to find the document again.

Searching the Web

Introduction

The web is a very eclectic and quite haphazardly organised collection of data. Unlike a library, which has a numeric system of classifying books by subject areas that facilitates fairly easy retrieval of a particular book, the web has no universal method of organising data. Given a particular topic that you might want to find information about there are two important ways of approaching the task. One way is to conduct a search of the web using one of the numerous search engines that are available, the most popular being Google.

There are many very useful tutorials on the World Wide Web that can help you to learn about searching. One very good one can be found at
www.sc.edu/beaufort/library/pages/bones/bones.shtml.
In addition to reading this section of the lab you should read all of this on-line tutorial.

Simple Searching with Google

There are many search engines available for the web. Most of these work by searching an index that has been compiled from "all" of the documents on the web. The word "all" is in quotes because it is never actually literally all documents (why?). The index can be thought of as a dictionary in which the entry for each word contains a list of the URLs of the web documents that contain the word. Of course words like of, the, and, at, etc. are not included in the dictionary. The compilation of the index is ongoing - that is parts of the index are being continually updated. But when you conduct a search you are actually searching an index that is mostly already out of date.

You'll find then that searches using different search engines will produce different results. The fact that the indexes are always being updated is one reason for this. Other reasons arise from the fact that the creation of the index may be restricted in some way (perhaps to only part of the whole collection of web documents) and the fact that different search engines use different strategies for trying to match your search specification.

To go to the Google web site type www.google.ca as the URL (either in the Location text box or in the Open Page … window).

There should be a Help link (About Google) somewhere in the document, which you should now click on. ***Read the Help information carefully - there's quite a lot to read - in order to understand the use of capital letters, quotes ("…."), the + sign and the - sign, the * symbol and other details of specifying a search query.*** Return to the main Google page when you're ready to try a search.

Creating Your Own Web Documents

Introduction

The purpose of this section is to further your understanding of how the WWW works by having you create a simple document that can be viewed using a WWW browser. We will not introduce you to the full scope of multimedia authoring for the web - this section will just skim the surface - but it should nevertheless remove much of the mystery surrounding what you have seen. The document you create will not be visible to anyone else using the WWW since your computer does not have the necessary WWW server software. The document will only be accessible to you though the WWW client browser, but it will contain links that take you out to other documents on the web.

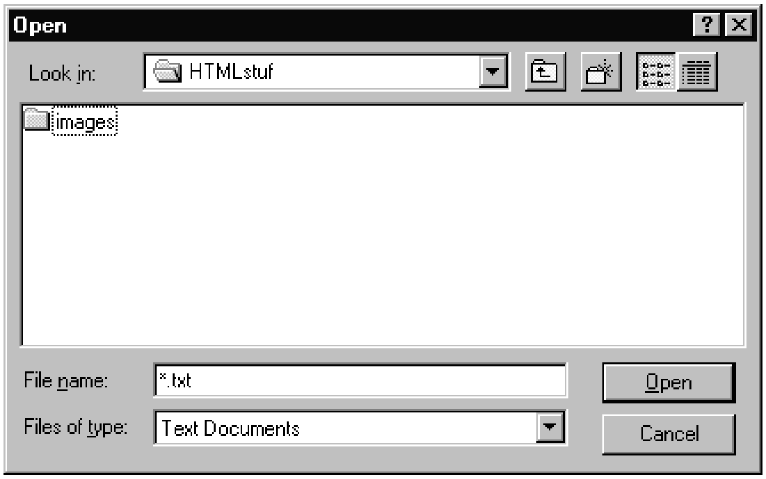
Documents that are intended to be displayed by a WWW browser must contain instructions telling the WWW browser how to format the display. These instructions are called "mark-up" instructions and the set of instructions make up a language. For WWW documents the language is called HTML - for hypertext markup language. An HTML document consists of the text that you actually want to appear as well as these instructions telling the WWW browser program how to format the text, which of course you don't want to appear. The formatting commands make up only a small part of HTML, however. There are also commands instructing the WWW browser to insert images into the document, or to insert sound or video, or to link to a different document, a specific place in a different document, or a different place in the same document. There are many other kinds of instructions as well, although the ones mentioned above are about all we will tackle here.

The file containing the text and markup instructions must be named with the .htm (or .html on computers other than Windows/PC machines) extension, which causes the WWW browser to recognise it as a file that the browser can display. This .htm file, although created by a person, is not really intended for a person to read. Because of the html markup instructions it is really meant for the WWW browser program to read and consequently the file looks rather odd.

Locate HTMLstuf in Support Files (Chapter 9) on the course website. It contains a file called boPeep.htm and folders called images and rhyme. Copy the whole HTMLstuf folder to your own personal folder and then double click on the boPeep.htm file. Because of the .htm extension your default web browser, probably Internet Explorer, launches and you see the document displayed. Note that you have not used the Internet to obtain this document - the file is on the Glade server.

Leaving the web browser window open, from the Start button choose Programs, Accessories and then Notepad. Notepad is a very simple editor that can do no formatting and

Figure 2.1 the Open panel of Notepad


Look in: HTMLstuf

images

File name: *.txt

Files of type: Text Documents

therefore displays only ASCII characters. Choose File/Open from the Notepad command menu and navigate through the file system to the HTMLstuf folder (Figure 2.1) now in your own personal folder. Except for the images and rhyme folders it will appear to be empty because Notepad expects to open only files with the .txt extension and there are none in the HTMLstuf folder. Choose All Files from the Files of type: pulldown menu and you should see the full list of files in the HTMLstuf folder. Choose the boPeep.htm file and click on Open.

You can now see the contents of the .htm file as the WWW browser sees it - i.e. with the html markup instructions visible. We'll examine this file in detail and modify it a bit.

Elements of HTML

Probably the first thing you notice about this file are the many words enclosed in angle brackets < ... >. These things are all HTML instructions. You'll notice that these things do not appear when the web browser displays the file.

The first few lines of the file are as follows:

```
<HTML>
<HEAD>
  <TITLE>My Profile</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>
Home Page for Little Bo Peep</H1></CENTER>
<H2>
About Me</H2>
```

The first tab <HTML> tells the WWW browser that what follows is html. At the very bottom of the file there is another tag </HTML> indicating the end of the html. You can also see <HEAD> … </HEAD>, <TITLE> … </TITLE>, <CENTER> … </CENTER>, and <H1> … </H1> tags. This use of pairs of instructions, the second of the pair including a / symbol, is a feature of the HTML language. Characters contained in < … > brackets are not displayed by the WWW browser, they are instructions rather than data to be displayed.

In these first few lines the only data actually displayed by a WWW browser are the characters My Profile, Home Page for Little Bo Peep, and About Me. The web browser window should still be on your screen - click in it to bring it to the front and observe where this data is displayed. You might miss My Profile - it's in the title bar of the window, but the other two lines of text are obvious.

In HTML different levels of headings are enclosed by <Hx> … </Hx> tags where x may be 1, 2, 3, 4 etc. Notice the different sizes of the headings H1 and H2. Change the <H1> and </H1> tags to <H6> and </H6> and then save the file using Save (not Save As) from the File menu of Notepad. Then go to the web browser window and click on the Reload button. Change the heading specifiers back again after seeing the result of this.

Next you see the line

```
<IMG SRC="images/sheep.jpg" HEIGHT=380 WIDTH=250 ALIGN=RIGHT>
```

This is a way that images can be included in documents. Notice first that this tag does not have a matching </IMG>. The piece SRC="images/sheep.jpg" specifies the location and filename of a file that contains the image to be displayed. The directory path is specified as images/ and the filename is sheep.jpg. Look in the images folder that was part of the HTMLstuf folder you copied to your personal directory previously. One of the files there is named sheep.jpg. The HEIGHT and WIDTH specifiers are not strictly necessary in our example, but lets see what the ALIGN=RIGHT specifier means. Change it to ALIGN=LEFT, save the file, go to the web browser, and reload the document to see the effect of this change.

The next part of the HTML document is some text. The <P> tags embedded in this text indicate where a new paragraph is to begin. Observe that blank lines or excess spaces between words have no effect on how the WWW browser displays the text. They are simply ignored. Its not easy to read the text with these spaces and blank lines so remove them now. Leave the blank lines before each new paragraph tag because these are helpful for us (although ignored by the WWW browser program).

The <BR> tag enforces a line break - it represents pressing the Enter key. To understand the <BR> tag first go to the web browser window and change its size - make it narrower for example. You'll see that the text is adjusted to wrap onto new lines in order to fit within the window. Now return to the Notepad window and type <BR> as shown:

```
              <P>I spend most of my days looking <BR> after my father's
```

Save the file, return to the web browser, reload the file and now try changing the window size. Notice that the words beginning after my father's … always begin on a new line.

The <HR WIDTH="100%"> tag creates the horizontal line you see in the web browser window just after the line One day I lost my father's sheep. Change the 100% to 50%, save the file and reload the document in the web browser to see the effect.

Links to Other Documents

The rest of this document doesn't contain any new features. However, Bo Peep would like to add to it references to other documents about herself on the web. So, following the last title, Web Sites About Me, add the following line of text:

```
   <A HREF="http://www.ehow.com/how_2085724_make-little-bo-peep-costume.html">
   How to make your own Bo peep costume</A>
```

Save the file, reload in the web browser, and you should see in underlined blue text, indicating a link, the words

How to make your own Bo peep costume

Click on the link to see the document. This document is out there on the Internet rather than being a local file such as we have been using until now.

Bo Peep has other references to add to this list of web sites, and fortunately HTML provides a nice way of creating a list via the <UL> … </UL> or <OL> …. </OL> tags. The former creates a list with bullets before each item and the later creates a list in which each item is numbered.

So type the <UL> tag on the line before the <A HREF= … line you previously wrote, and place the tag <LI>, meaning list item, in front of <A HREF=… Then add a second web link as shown next:

```
<UL>
 <LI><A HREF="http://www.ehow.com/how_2085724_make-little-bo-peep-costume.html">
 How to make your own Bo peep costume</A>
<LI><A HREF="http://www.toywiz.com/toystoryexclusivebopeep.html">
Collectable Figurine</a>
</UL>
```

Locate some more references to Bo Peep on the Web and include a few more in this list.

The Little Bo Peep nursery rhyme is perhaps not known to everyone - so Bo Peep would like to include it, not as part of her home page but as a separate file which can be accessed via a link connected to her name in the very first heading. You'll find the nursery rhyme in the file rhyme.txt in the rhyme folder of the HTMLstuf folder. Double click on it and it should open in Notepad (because it has the .txt extension).

Your task is to add the necessary markup tags to make this an HTML file - use the boPeep.htm file as a model. When you have done this use Save As … to save the file with the name rhyme.htm (note the different extension).

Now use Notepad to edit the boPeep.htm file in order to make the link to the rhyme.htm file. Near the top of the file are the following lines

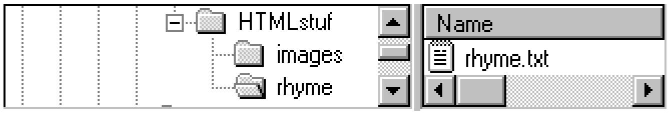
```
<CENTER>
<H1>
Home Page for Little Bo Peep</H1></CENTER>
```

and you would like the words Little Bo Peep to be a link to the rhyme.htm file. You need to modify this text as follows:

```
<CENTER>
<H1>
Home Page for <A HREF="rhyme/rhyme.htm">Little Bo Peep</A>
</H1>
</CENTER>
```

Notice how the link to the file, specified by HREF="….." includes the directory path rhyme/ as well as the file name rhyme.htm. The directory path as specified here is relative to the position of the file that contains the link, i.e. the boPeep.htm file.

Bo Peep decides she would like to include a picture in the nursery rhyme document. There is an image - called bopeep-pic.jpg - in the images folder, and you should now add the necessary link to the rhyme.htm file which you just modified. The file containing the image is in a different folder however and you need to be careful specifying the directory path. You cannot use SRC="images/bopeep-pic.jpg" because there is no images folder in the rhyme folder (which is where the rhyme.htm file is located). The directory structure looks as follows:



The bopeep-pic.jpg file is found by backing up one level to the HTMLstuf folder and then going into the images folder. You specify this as SRC="../images/bopeep-pic.jpg", where the ../ represents backing up one level in the directory structure.