

Introduction to Web Technologies

Tara Murphy

17th February, 2011

The Internet is a network of networks

- The Internet is the descendant of ARPANET (Advanced Research Projects Agency Network) developed for the US DoD
- The initial goal was to research the possibility of remote communication between machines
- Critical step was development of the TCP/IP protocol (1977)
 - TCP Transmission Control Protocol
 - IP Internet Protocol
- Vinton Cerf's postcard analogy for TCP/IP:
 - A document is broken up into postcard-sized chunks (packets)
 - Each postcard has its own address and sequence number
 - Each postcard travels independently to the final destination
 - The document is reconstructed by ordering the postcards
 - If one is missing, the recipient can request for it to be resent
 - If a post-office is closed the postcard is sent a different way
 - Congestion and service interruptions do not stop transmission

The first connection between two hosts

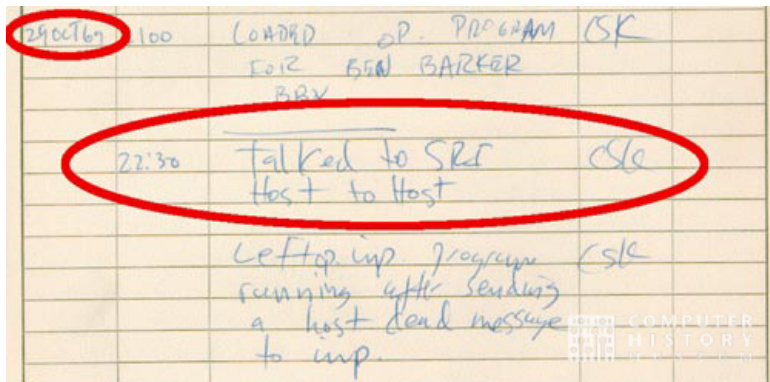
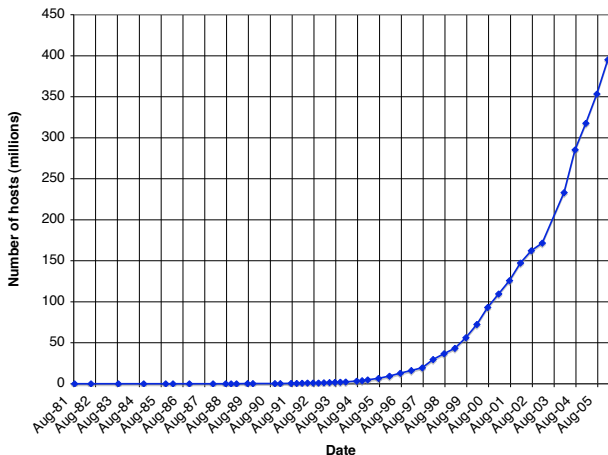


Image Ref: <http://www.computerhistory.org>

The Internet grew extremely rapidly!



Data Ref: <http://www.isc.org/>



The World Wide Web operates over the Internet

- We often use the phrases “*the WWW*” and “*the Internet*” interchangeably, however they are different entities
- The WWW is a *service* that operates over the internet
- The concept of the WWW combines 4 ideas:
 - hypertext
 - resource identifiers (URI, URL)
 - client-server model of computing (web servers/browsers)
 - markup language (HTML)
- These were the brainchild of Tim Berners-Lee from CERN who released his first browser in 1991
- All clients and servers in the WWW speak the language of HTTP (HyperText Transfer Protocol)

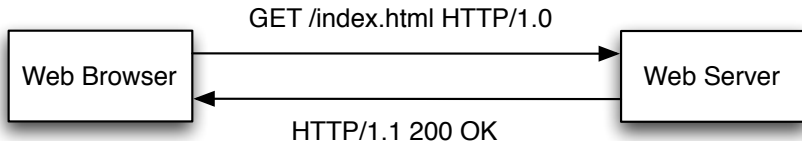
We can *generate* content dynamically

- There are several benefits to dynamically generating content:
 - We don't have to store loads of pages
 - The content is completely up-to-date
 - **We can respond to/interact with the user**
 - Every site that involves a transaction (eg. Google, Amazon, NED) is generating dynamic content

Web servers *serve content on request* across the network

- The web server is responsible for:
 - accepting requests for content described by the URL
 - checking whether access is permitted and requesting authentication if necessary
 - sending (or *servng*) the content back to the browser
- A *web server* is the machine *and* the process serving content
- The most popular web server software now is:
 - Apache is an open source web server (Unix/Mac OS X/Win)
 - Microsoft IIS is the main Windows web server (Win only)

Browsers and servers communicate via HTTP



- *HyperText Transfer Protocol* (HTTP) is the standard protocol for transferring web content
- The server *listens* on port 80 waiting for connections
- The web browser connects to the server, and sends a *request*
- The server *responds* with an error code or the web content

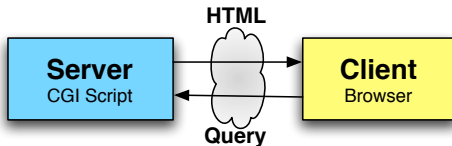
The server runs a program to generate the web content

- This program gets run *every time* the given URL is requested
- The server passes the HTTP request details to the program
- The program returns the web content or an error code
- Each web server interacts with the programs differently:
 - Apache uses the *Common Gateway Interface* (CGI)
 - Microsoft IIS uses *Active Server Pages* (ASP)

Browsing the web uses the client-server model

- The *client-server model* involves networked interaction between:
 - a *client* – in this case the web browser
 - a *server* – in this case the web server
- Dynamic content is generated on the *server side*
- The advantages of server side are:
 - We are not running programs on low-powered client computers
 - Typically the data you want to present is on server side
 - The client will restrict program functionality for security
- The disadvantage of server side are:
 - The server requires lots of processing power *particularly when there are many simultaneous clients*
 - The client side is often quite powerful anyway
 - Lots of information may need to be passed back and forth

The CGI client-server interaction



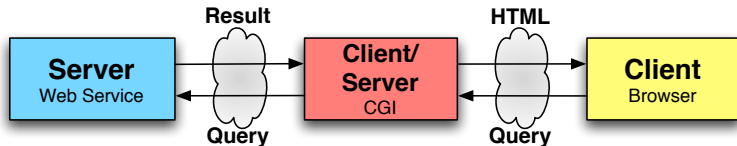
A web service is an application accessible over the Internet

- Web services emerged amidst a lot of hype
- A web service is a network accessible interface to application functionality, built using standard internet technologies.
- Powerful new way to build software systems from distributed components
- In other words, if an *application* can be *accessed over a network* using protocols such as HTTP, XML, SMTP etc. then it is a web service.

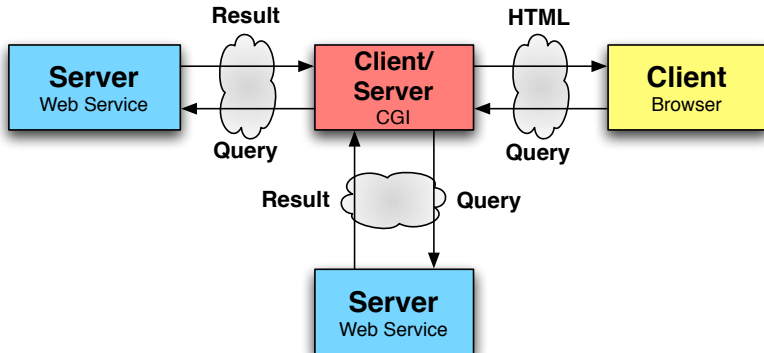
Web services use the client-server model

- Recall the CGI client-server model
- In the case of a user looking at a webpage
 - the *client* is the web browser
 - the *server* is the web server (and programs running on it)
- On the WWW information is always returned to the client in the form of a webpage (HTML).
- The key to web services is that they return information in a programmatic form (ie: they can return a string, float, array, object, just like an function).
- In the final stage of a chain of web services, the information may be presented to the user e.g. a webpage may be generated.

The Web service-client interaction



The Web service-client interaction



Example web services

- Stock price quotes
- Amazon web services
 - provides access to the entire Amazon database of books/prices
 - you could aggregate prices for multiple online bookshops
- Google web services
 - originally just access to Google search engine results
 - people used to do this *manually* anyway – screen scraping
 - now extended to other services, e.g. Google maps
- And lots of astronomy/VO applications
 - Andreas will show some examples this afternoon

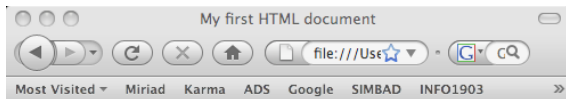
The HyperText Markup Language

- HTML marks up the structure of a document for publishing on the WWW
 - It tells the browser how to interpret and display the document
 - Different browsers interpret things differently (!)
-
- There are two main standards: HTML 4(5) and XHTML 1.0
 - These are developed by W3C
 - W3C the World Wide Web Consortium
 - All HTML documents should declare which standard they are using

Hello world!

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
2   "http://www.w3.org/TR/html4/strict.dtd">
3 <HTML>
4   <HEAD>
5     <TITLE>My first HTML document</TITLE>
6   </HEAD>
7   <BODY>
8     <P>Hello world!
9   </BODY>
10 </HTML>
```

Hello world!



Hello world!



The basic unit of HTML is the *element*

- HTML includes element types to represent paragraphs, hypertext links, lists, tables, images, etc
- Each element consists of three parts
 - ① start tag e.g. `<title>`
 - ② content e.g. my homepage
 - ③ end tag e.g. `</title>`
- A *tag* is an element name enclosed in angle brackets
- Some elements have no content e.g. `
` or `<hr>`
- Elements may have associated properties (*attributes*)
- Attributes and their values appear inside the start tag e.g. `<div id="section1">`

You only need a small set of elements to create a website

Element: start/end tags	Description
<code><html> </html></code>	Starts and ends a HTML document
<code><title> </title></code>	Text that appears in the title bar
<code><head> </head></code>	Information about the document
<code><body> </body></code>	The main part of the document
<code><p> </p></code>	A paragraph
<code><hr /></code>	A horizontal line
<code>
</code>	A line break
<code> </code>	A link
<code></code>	An image
<code><!-- comment --></code>	Comments that are not displayed

You only need a small set of elements to create a website

Element: start/end tags	Description
<code><div> </div></code>	A section in the document
<code> </code>	An inline section in a document
<code> </code>	An unordered list (bullet points)
<code> </code>	An ordered list
<code> </code>	A list item
<code><table> </table></code>	Encloses a table
<code><tr> </tr></code>	A row in a table
<code><td> </td></code>	A cell within a row
<code><pre> </pre></code>	Enclosed text that stays in its raw format

CSS was introduced into HTML 4.0 to solve a problem

- We have focused on the structural aspects of HTML
- In fact that is what HTML was originally designed for
 - `<table>` = “This is a table”
 - `<p>` = “This is a paragraph”
- Layout was the job of the browser
- As the WWW exploded, more people started writing documents
- The two major browsers (Internet Explorer and Netscape) added new HTML tags and attributes to the original HTML specification e.g. ``
- It became hard to separate *structure* and *presentation*

Formatting before CSS was inefficient

- Before CSS all formatting had to be included as attributes in HTML tags

```
1 <font face="Verdana, Arial" size="+1" color="blue">  
2   Hello, World!  
3 </font>
```

- There are several disadvantages to this way of doing things
 - Information occurs in many locations → redundancy → errors
 - Updating multiple occurrences of information is time-consuming
 - Formatting information is hard-coded in HTML document
 - HTML elements can describe format/presentation and content/structure
- Other formatting tags you might be familiar with a `` (bold), `<i>` (italics). . . **we do not recommend using these**

Hello World! the CSS version

- To reproduce the previous HTML using CSS we need two files
 - A HTML page (e.g. mypage.html) containing this

```
1 <head>
2   <link href="css/mystyle.css" rel="stylesheet"
3       type="text/css" />
4 </head>
5 <body>
6   <p>Hello, World!</p>
7 </body>
```

- An accompanying style sheet file (e.g. mystyle.css)

```
1 p {
2   color: blue;
3   font-size: small;
4   font-family: Verdana, Arial, sans-serif;
5 }
```

HTML and CSS should be validated

- The W3C site provides tools for *validating* your website
 - they check what standard you *claim* to be using
 - then check all the syntax in your document complies with that standard
- The validators are free and easy to use, so there is no excuse!
- <http://validator.w3.org/>
- <http://jigsaw.w3.org/css-validator/>



References

- <http://www.computerhistory.org>
- <http://www.anu.edu.au/people/Roger.Clarke/II/OzIHist.html>
- HTML: <http://www.w3.org/MarkUp/>
- HTML: <http://www.w3schools.com/html/>
- XHTML: <http://www.w3.org/MarkUp/2004/xhtml1-faq>
- XHTML:
http://www.w3schools.com/xhtml/xhtml1_html.asp
- CSS: <http://www.w3.org/Style/CSS/>
- CSS: <http://www.csszengarden.com/>