The World Wide Web

Outline

- Background
- Structure
- Protocols

WWW Background

- 1989-1990 – Tim Berners-Lee invents the World Wide Web at CERN
  - Means for transferring text and graphics simultaneously
  - Client/Server data transfer protocol
    - Communication via application level protocol
    - System ran on top of standard networking infrastructure
  - Text mark up language
    - Not invented by Bernes-Lee
    - Simple and easy to use
    - Requires a client application to render text/graphics

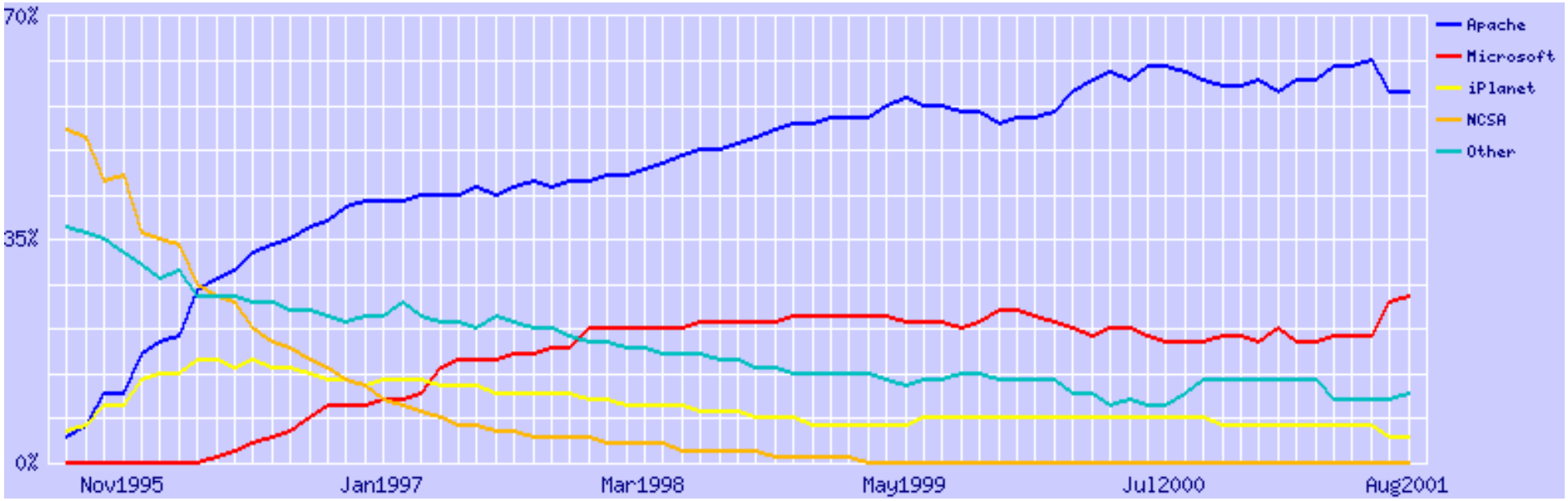
WWW History contd.

- 1994 – Mark Andreesen invents MOSAIC at National Center for Super Computing Applications (NCSA)
  - First graphical browser
  - Internet's first "killer app"
  - Freely distributed
  - Became Netscape Inc.
- 1995 (approx.) – Web traffic becomes dominant
  - Exponential growth
  - E-commerce
  - Web infrastructure companies
  - World Wide Web Consortium
- Reference: "Web Protocols and Practice", Krishnamurthy and Rexford

WWW Components

- Structural Components
  - Clients/browsers – to dominant implementations
  - Servers – run on sophisticated hardware
  - Caches – many interesting implementations
  - Internet – the global infrastructure which facilitates data transfer
- Semantic Components
  - Hyper Text Transfer Protocol (HTTP)
  - Hyper Text Markup Language (HTML)
    - eXtensible Markup Language (XML)
  - Uniform Resource Identifiers (URIs)

Quick Aside – Web server use

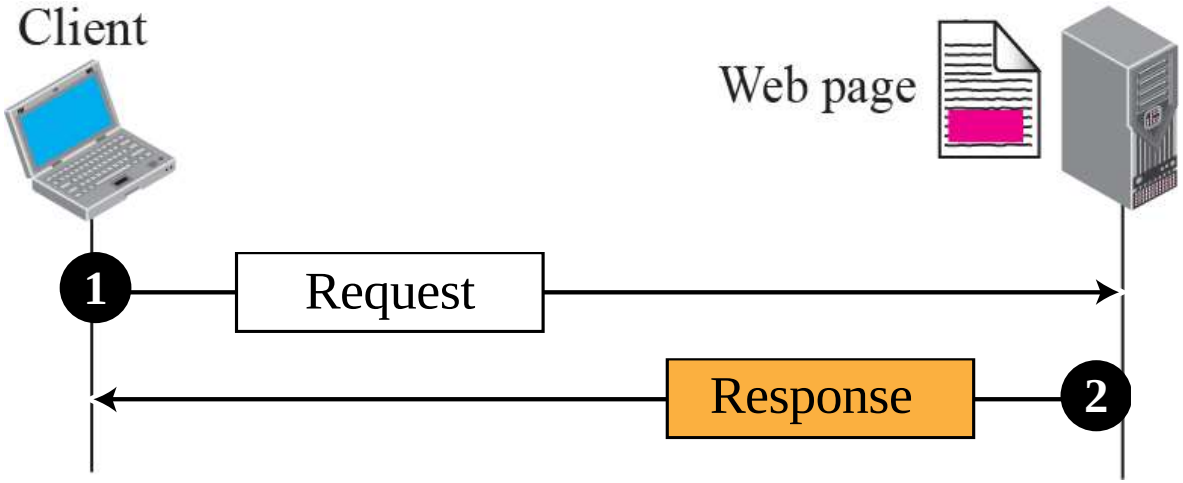


Source: Netcraft Server Survey, 2001

WWW Structure

- Clients use browser application to send URIs via HTTP to servers requesting a Web page
- Web pages constructed using HTML (or other markup language) and consist of text, graphics, sounds plus embedded files
- Servers (or caches) respond with requested Web page
  - Or with error message
- Client's browser renders Web page returned by server
  - Page is written using Hyper Text Markup Language (HTML)
  - Displaying text, graphics and sound in browser
  - Writing data as well
- The entire system runs over standard networking protocols (TCP/IP, DNS,…)

Figure 22.1 *Example 22.1*



What happens when you type a URL in the browser and press enter?

1. **URL is converted to IP address**

2. **The browser initiates a TCP connection with the server**.

3. **Data exchange happens using HTTP**

4. **Page is rendered in web browsers**

e.g. http://example.org/home.html

1. IP address such as *203.0.113.4*

2. It requests service from a specific TCP port number (80) that is well known for the HTTP service

3. The content of HTTP request can be simple two lines

GET /home.html HTTP/1.1
Host: example.org
The computer receiving the HTTP request delivers it to web server software listening for requests on port 80. If the web server can fulfil the request it sends an HTTP response back to the browser indicating success:

HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8

followed by the content of the requested page. Hypertext Markup Language (HTML) for a basic web page might look like this:

```
<html>
  <head>
    <title>Example.org – The World Wide Web</title>
  </head>
  <body>
    <p>The World Wide Web, abbreviated as WWW and commonly known
...</p>
  </body>
</html>
```

As it receives their content from the web server, the browser progressively renders the page onto the screen as specified by its HTML and these additional resources.

Uniform Resource Identifiers

- Web resources need names/identifiers – Uniform Resource Identifiers (URIs)
  - Resource can reside anywhere on the Internet
- URIs are a somewhat abstract notion
  - A pointer to a resource to which request methods can be applied to generate potentially different responses
    - A request method is eg. fetching or changing the object
- Instance: http://www.foo.com/index.html
  - Protocol, server, resource
- Most popular form of a URI is the Uniform Resource Locator (URL)
  - Differences between URI and URL are beyond scope
  - RFC 2396

Figure 22.4 *URL*

Protocol can be http, ftp,smtp,telnet