World Wide Web: URL, HTML, HTTP

World Wide Web (WWW)

- created in 1989 at CERN by Tim Berners-Lee
- ideas (hypertext, information store) are much older
- linked retrieval of text, images, sounds, movies, …
- also executable programs (Java)
- predecessor: gopher
- hypertext = linking objects (text) across a network
- integrates data access protocols
- graphical browser ➔ Internet for non-techies
- currently most important Internet application (next to email)
WWW components

names: Universal Resource Locator (URL)
retrieval: Hypertext transfer protocol (HTTP)
hypertext: Hypertext markup language (HTML)
browser: retrieve URL via HTTP, ftp, telnet, ... from server
server: stores objects for one or more host names

URL

scheme://host[:port]/path/[;parameters][#fragment][?query]

http://www.fokus.gmd.de:80/step/hgs/
mailto:president@whitehouse.gov

scheme: http, ftp, gopher, telnet, mailto, news, ...
host: host name or dotted quad
port: TCP port, if not default
path: file name path
parameters: for partial retrieval, etc.
fragment: named position in text
query: parameter for database queries
HTML

- simple document type for SGML (standard general markup language)
- describes document structure, not formatting (kind of…)
- provision for hyperlinks
- can be …
  - written by hand
  - generated by special authoring tools
  - translated from \LaTeX, nroff, Microsoft Word, …

HTML structure

- `<tag parameter=value ...> text </tag>`
- uses ISO 8859-1 (including ä, ö, ü, etc.)
- can be searched locally (↔ PostScript)
- display reasonably on wide range of hardware (ASCII terminal, different resolutions, braille, audio output, …)
- has tables, but currently no mathematics
- some standardized (HTML 2.0), much vendor-introduced (Netscape)
- constantly evolving
**HTML document structure**

```html
<HTML>
<HEAD>
<TITLE>Title of document</TITLE>
</HEAD>
<BODY>
The body of the document
</BODY>
</HTML>
```

**HTML tags**

- headings: H1 ... H6
- paragraph: P
- preformatted text: PRE
- quote: BLOCKQUOTE
- unordered list:
  ```html
  <UL>
  <LI>first list item
  <LI>second list item
  </UL>
  ```
- ordered list
<OL>
  <LI>first list item</LI>
  <LI>second list item</LI>
</OL>

- definition list

<DL>
  <DT>HTML</DT>
  <DD>Hypertext markup language</DD>
  <DT>HTTP</DT>
  <DD>Hypertext transport protocol</DD>
</DL>

**HTML phrase markup**

- typographic elements ("how to show this")
  - **bold**: `<B>bold</B>`
  - *italic*: `<I>italic</I>`
  - `teletype`: `<tt>text</tt>`
  - blinking text: `<blink>avoid!</blink>`

- idiomatic elements ("what is this")
  - citation: `<cite>`
  - emphasis: `<em>`
  - strong emphasis: `<strong>`
  - variable: `<var>`
  - ...
**HTML miscellaneous**

- horizontal rule: `<HR>`
- line break: `<BR>`
- automatic playback of sounds
- backgrounds
- frames
- font size changes
- ...

**HTML forms**

- allows user to input text, select choices, ...
- encoded as `name1=value1&name2=value2`
- escape sequences for space (+) and non-alphanumeric characters
- returned as GET or POST (explained later)

```html
<FORM METHOD="POST" ACTION="http://www.w3.org/sample">
Your name: <INPUT NAME="name" SIZE="48"><p>
Male: <INPUT NAME="gender" TYPE=RADIO VALUE="male"><p>
Female: <INPUT NAME="gender" TYPE=RADIO VALUE="female"><p>
Cities in which you maintain a residence:<p>
<ul>
<li>Kent <INPUT NAME="city" TYPE="checkbox" VALUE="kent"> <p>
<li>Miami <INPUT NAME="city" TYPE="checkbox" VALUE="miami">
</ul><p>
<INPUT TYPE=SUBMIT> <INPUT TYPE=RESET></FORM>
```
HTML links

- source: `<a href="url">word</a>` ➤ usually shown underlined and blue, followed when clicked on
- destination within document (#fragment):
  `<a name="frag">word</a>` ➤ not visible to user
- `<IMG SRC="jim.jpg" ALT="Jim’s picture">` displays GIF, JPEG concurrently, without user intervention

HTTP

- simple ASCII-based transfer protocol
- methods:
  - **GET**: retrieve object, no side effects
  - **HEAD**: get descriptive information
  - **PUT**: replace object on server
  - **others**: PATCH, COPY, MOVE, …
- stateless ➤ every query independent
- one request ➤ one TCP connection (client opens, server closes) ➤ inefficient ➤ HTTP 1.1 connections
- uses ftp-style error codes
HTTP request

GET /index.html HTTP/1.0
Accept: text/html, */*
Referer: http://www.w3.org/hypertext/
From: webmaster@w3.org

(empty line marks end of request)

HTTP response

HTTP/1.0 200 Document follows
Date: Wed, 31 Jan 1996 20:45:17 GMT
Server: NCSA/1.5
Content-type: text/html
Content-language: en
Last-Modified: Wed, 31 Jan 1996 20:00:10 GMT

<html>
<head>
...
</html>

(server closes connection)
HTTP access control

- client attempts access (GET, PUT, ...) normally
- server returns

HTTP/1.0 401 Unauthorized
WWW-Authenticate: Basic realm="WallyWorld"

- client tries again

Authorization: Basic base64-encoded-user:password

- passwords in the clear ➔ not secure
- ➔ use challenge-response instead (using shared secret)

Caching and proxies

- avoid repeated transfers ➔ cache object in memory, disk or on a proxy
- proxy:
  - only for HTTP
  - receive normal HTTP request
  - check if available locally and still current (HEAD If-modified-since)
  - if not, retrieve, return to client and store
- also used for firewalls
Problems

- locations may change ➙ “dangling” pointers ➙ indirection (URNs)
- little descriptive information ➙ URC
- no privacy, strong authentication ➙ SSL (generic TCP), SHTTP
- finding objects ➙ directories, “spiders”
- caching, proxies ↔ charging, counting “hits”