1.264 Lecture 16

Web introduction, HTTP

Next class: ASP.NET book chapters 2, 3. Exercises due after class

Exercise 1: HTTP request

Example browser request: (Browse to athena.dialup.mit.edu, log in)

```
telnet web.mit.edu 80       (telnet opens connection, http is port 80)
GET /admissions/   HTTP/1.1                         (request)
Host: web.mit.edu                                   (header)
(blank line)
```

Example server response: (Hand in screen shot of response)

```
HTTP/1.1 200 OK                                     (response)
Server: Apache/1.3.3 Ben-SSL/1.28 (Unix)            (header)
Content-Type: text/html
Content-Length: 8300
(blank line)
<HTML>                                              (MIME (html) document)
<HEAD><TITLE>MIT…</TITLE></HEAD>
<BODY>MIT admissions…</BODY>
</HTML>
Connection closed                                   (close connection)
```

(athena.dialup.mit.edu)   Ctrl-] to esc

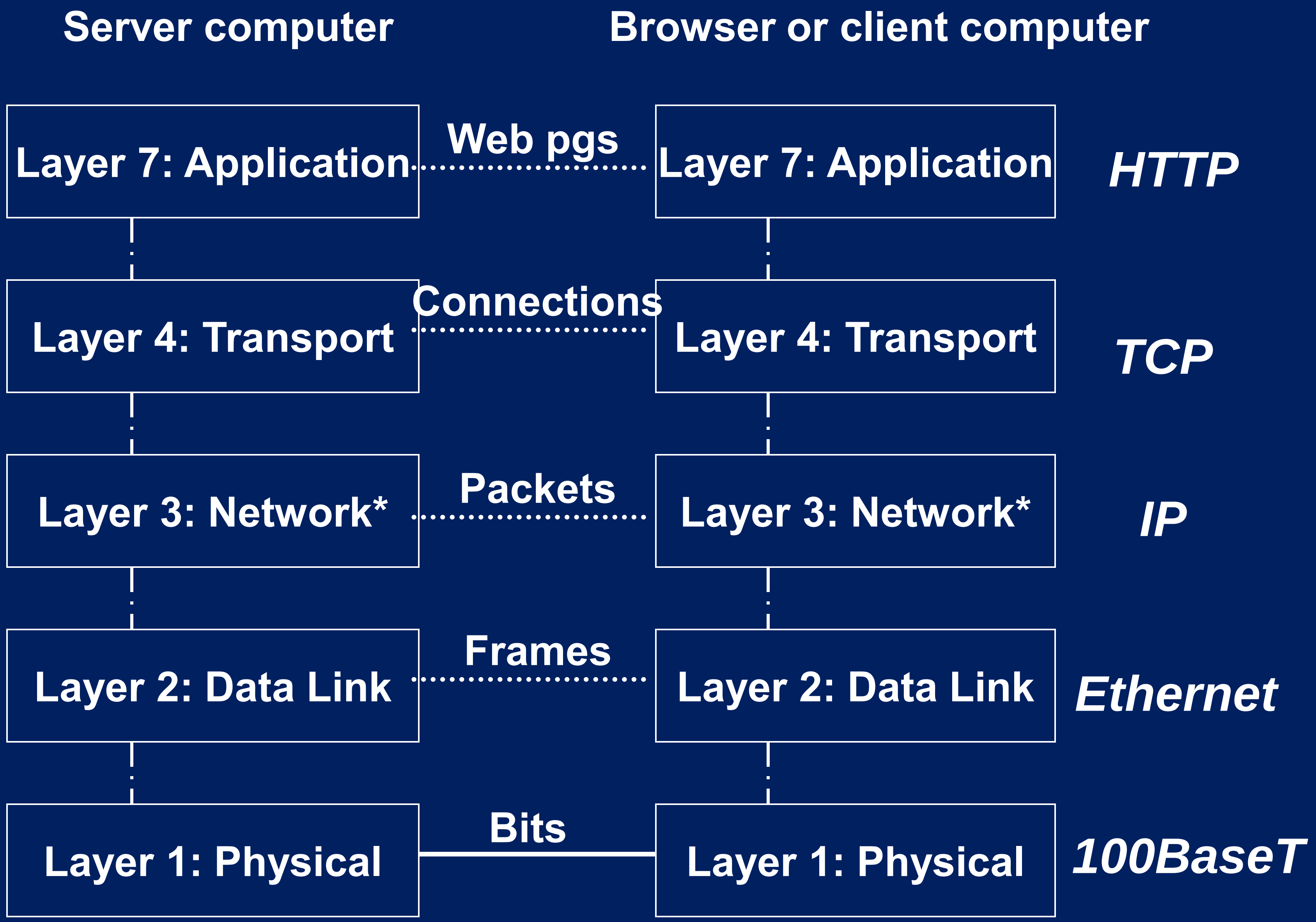
Internet and Web Context

- We have covered databases, which are the <u>nodes</u> in our information network where data is stored
- Internet and Web are the <u>links</u> that connect the nodes to transfer data from one node to another
  - Web is one of many services that run over the Internet
  - Internet is system of interconnected computer networks
  - Network has protocols to exchange data
- Web is a system of hyperlinked documents
  - Accessed using the Internet,
  - Referenced by Uniform Resource Locators (URLs)
- Web, for business, is a communications channel to transfer data from one database to another
  - Documents can be dynamically generated, validated (using business rules) and exchanged
  - Documents are often fragments of a database

Internet and TCP/IP

- Internet is a set of loosely interconnected networks
  - Local area networks connected via wide area networks
  - No centralized control or management
  - Network segments interconnect via <u>routers:</u>
    - Dedicated computers that manage packets of data
  - TCP/IP is the universal <u>data</u> protocol on the network
    - TCP/IP: Transport control protocol/Internet protocol
    - <u>Content</u> is left to higher-level protocols, like the Web
    - <u>Transport</u> left to lower-level protocols, like Ethernet/WiFi
- TCP/IP connections are central to Internet and Web
  - Client is typically a data <u>consumer</u>
    - On Web, client is a browser or an application program
  - Server is typically a data <u>provider</u>
    - Web server listens for requests and transmits desired data
    - Web server gets data from application or database server
  - Web connection is active only long enough to exchange data
    - Avoid connection overhead, but lose session state

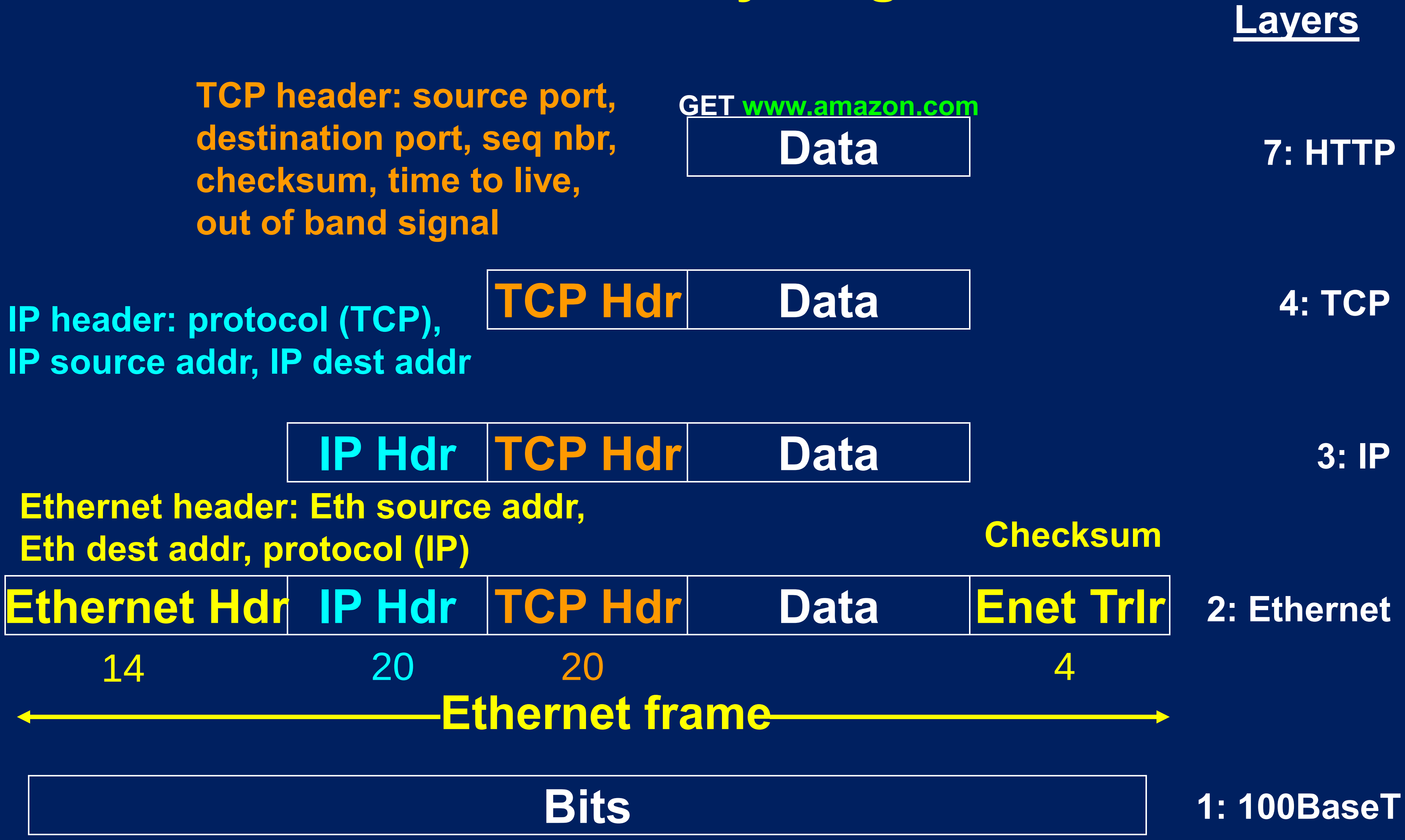
Transmission Control Protocol/Internet Protocol TCP/IP



Server computer		Browser or client computer	
Layer 7: Application	Web pgs	Layer 7: Application	*HTTP*
Layer 4: Transport	Connections	Layer 4: Transport	*TCP*
Layer 3: Network*	Packets	Layer 3: Network*	*IP*
Layer 2: Data Link	Frames	Layer 2: Data Link	*Ethernet*
Layer 1: Physical	Bits	Layer 1: Physical	*100BaseT*

*-not same as layer 1/2 “network”

TCP/IP layering



Layers

TCP header: source port, destination port, seq nbr, checksum, time to live, out of band signal

GET www.amazon.com

Data — 7: HTTP

TCP Hdr | Data — 4: TCP

IP header: protocol (TCP), IP source addr, IP dest addr

IP Hdr | TCP Hdr | Data — 3: IP

Ethernet header: Eth source addr, Eth dest addr, protocol (IP)

Checksum

Ethernet Hdr | IP Hdr | TCP Hdr | Data | Enet Trlr — 2: Ethernet

14 20 20 4

Ethernet frame

Bits — 1: 100BaseT

Exercise 2: Web, TCP/IP, Ethernet

- Your laptop browser at IP address 18.9.3.2 sends
  - Web (HTTP) command on port 80: GET amazon.com
    - Other ports: ftp: 21, ssh: 22, telnet: 23, smtp:25, DNS: 53
  - To amazon’s IP address: 72.21.214.128
  - As a single IP packet at max time-to-live (255 hops)
  - TCP checksum computed to be 5324
  - Laptop Ethernet address is AA:BB:CC:DD:EE:FF
  - Router Ethernet address is 00:11:22:33:44:55
  - Ethernet checksum computed to be 6435
- Draw the Ethernet frame as it leaves your laptop with all field values filled in
- Draw the TCP packet received at the amazon server with all field values filled in

Solution

- Ethernet frame sent from laptop:
  - AA:BB:CC:DD:EE:FF 00:11:22:33:44:55 IP TCP 18.9.3.2 72.21.214.128 80 80 0 5324 255 0 GET amazon.com 6435
- TCP packet at amazon server:
  - 80 80 0 5324 254 0 GET amazon.com
  - (Time to live may be less than 254)
  - (Source port may be different than 80)

Web organization: Web pages

- Web page or document is unit of organization
  - Document is basic unit of business interactions
  - Pages have hypertext
    - Display text and images
    - Links to other pages, implemented via embedded URLs
  - Pages are described using Extensible Hypertext Markup Language (XHTML)
    - High level document description language
    - Specifies layout but not exact appearance of document
    - Browser handles display of page on client machine
  - XHTML has replaced HTML
    - Subset of XML, more structured, verifiable than HTML
    - XML is eXtensible Markup Language

Web organization: MIME types

- Every document on Web has a type
  - Multipurpose Internet Mail Extensions (MIME) types, in http standard
    - Some types, such as html, text and some graphics, are displayed directly by browser
    - Others need helper or plug-in external programs to display type
  - Web type system is extensible. New types are easy to accommodate.
    - Define new MIME type in server
    - Browser will prompt for helper application if MIME type unrecognized
    - Browser can suggest Web site to download helper app
    - Download and install helper app, and use the new MIME type
  - Web servers often generate documents dynamically
    - URLs point at programs as well as pages
    - Programs written in Java Enterprise Edition (JEE6), Microsoft .NET Framework, or LAMP (Linux-Apache-MySQL-php)

MIME type examples

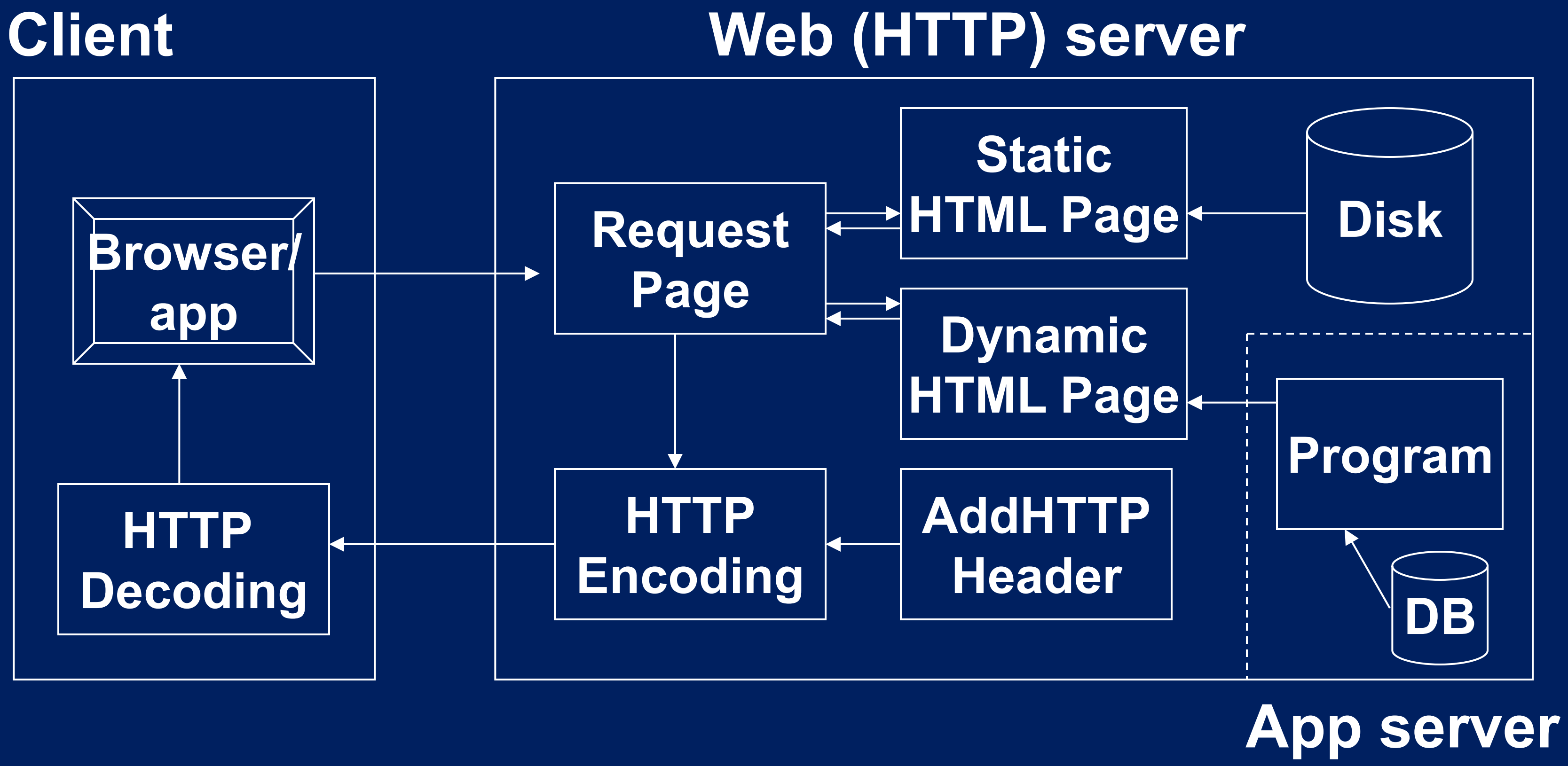
application/msword	Word
application/pdf	Acrobat
application/vnd.ms-excel	Excel
application/zip	Zip file
audio/basic	.au, .snd
image/gif	GIF
image/jpeg	JPEG
text/plain	Plain text
text/html	HTML
text/xml	XML
video/mpeg	Video

See http://www.rfc-editor.org for current list of HTTP, MIME, other Internet specifications

Hypertext transfer protocol: HTTP

- HTTP is core request-response protocol for Web
- Four phases:
  - Open connection: Based on URL
  - Request: Client opens connection to server and sends:
    - Request method, (and request data at bottom if POST or PUT request)
    - URL,
    - HTTP version number
    - Header information (informational, optional), terminated with blank line
  - Response: Server processes request and sends:
    - HTTP protocol version and status code
    - Header information, terminated by blank line
    - Text (data)
  - Close connection

HTTP



Client

Web (HTTP) server

Browser/ app

Request Page

Static HTML Page

Disk

Dynamic HTML Page

Program

DB

HTTP Decoding

HTTP Encoding

AddHTTP Header

App server

- These transactions are stateless. The connection is closed after each page and re-established: Server can’t connect successive requests from the client
- We use cookies and similar methods to simulate state

HTTP request phase (browser to server)

Command	Description
GET	Requests specified document (used to ‘post’ too)
HEAD	Requests only header of specified document
POST	Requests that server accept data from browser and generate dynamic content
OPTIONS	Get server and access options (OPTIONS *)
TRACE	Used in debugging
PUT	Replace server document with data from browser
DELETE	Delete specified document on server
CONNECT	Converts connection to tunnel, usually with SSL
PATCH	Applies partial modification to resource

- GET is the most frequent request from browsers
- POST used for processing forms (dynamic pages and input forms)
- HEAD used to check that content exists
- TRACE for debugging
- PUT, DELETE used for REST (Web service) middleware
- CONNECT used in secure connections

Common HTTP request headers

Header	Definition
Accept	MIME types accepted by client (multiple allowed)
Connection	Connection type for client (keep-alive, close)
Cookie	Return previous cookie to server (session data)
From	Email address of user (sent only by crawlers)
Host	Original host requested (forward, multiple names)
If-Modified-Since	Used to reduce fetching of docs client has
Referer	URL from which this link was obtained
User-Agent	Name and version of client software (browser)

HTTP response status codes

Status code	Definition
100-102	Informational response; client should respond with some other action (continue, new protocol)
200-207	Request was successful
300-307	Document has moved; indicate new address
400-499	Client error, such as unauthorized request
500-510	Server error

Examples:

200	OK
404	Not found
505	HTTP version not supported

HTTP response header examples

Header	Definition
Allow	Requests allowed, such as GET
Content-Length	Length in bytes of data to be returned
Content-Type	MIME type of returned data
Expires	Date at which document expires
Last-Modified	Date at which document was last modified
Location	New document address (with 300 status)
Set-Cookie	Gives browser a ‘cookie’

Glossary

- URL: Uniform Resource Locator (Internet)
- HTTP: Hypertext Transfer Protocol (Web)
- TCP: Transport Control Protocol (Internet layer 3)
- IP: Internet Protocol (Internet layer 4)
- 100BaseT: common Ethernet type (layer 2)
- HTML: Hypertext Markup Language (Web)
- XML: Extensible Markup Language (Web)
- XHTML: XML-based HTML (Web)
- MIME: Multipurpose Internet Mail Extensions
- JEE6: Java Enterprise Edition 6 (software)
- .NET: Microsoft software framework
- LAMP: Linux-Apache-MySQL-php (software)