Internet & Web

Dr. Reda Salama

CPIT405- Internet Programming

Your Assignment

- Your supervisor just handed you a spec for implementation of:
  - Distributed system with universal connectability using sockets
  - Can process an open-ended variety of documents
  - Expandable by 5 orders of magnitude in ten years
  - Can add new tools easily
  - Supports 50 million users a day without gridlock.
- You say ***NO WAY!***
- Well, maybe.

Introduction to Internet and Web

- This presentation addresses two questions:
  - Is that possible?
    - Well yes – look over there – the web!
  - How was it accomplished?
    - Processing structure and protocols (Http, HTML, TCP)
    - Programming tools
      - Web servers and browsers that host:
        - Script languages, e.g., Javascript, VBScript, Perl, Ruby, RMI, Servelet, JSP, JavaBeans, SOAP…
        - Programming languages:
          Visual Basic, Java, C++, C#, …
        - All in All some server-side & client-side technology
    - And, of course, some very smart people

Web Goals:

- Build distributed system to share documents.
  - Universal readership
    - When content is available it should be accessible from any type of computer, anywhere.
  - Interconnecting all things
    - Hypertext links everywhere.
    - Simple authoring
- Support expansion by 5 orders of magnitude in ten years – 200 hosts to 500 million hosts.
- Manage communication between hundreds of millions of machines every day without collapsing from congestion.
- Provide for arbitrary extensions:
  - From static text documents to graphics, dynamic content, streaming video, programmable interfaces, voice, …

The Internet and the WWW are Different

- The *Internet* is a global digital infrastructure that connects millions of computers and tens of millions of people

- The *World Wide Web* is a mechanism that unifies the retrieval and display of a subset of data on the Internet

- An *intranet* is a local/global information structure that connects an organization internally

Internet & Web Design Principles

INTERNET

- Goal is connectivity
- Achieved with Internet Protocol (IP)
  - Stateless so survives failures – no need to backup
- Made scalable with end-to-end intelligence
  - Transport Control Protocol (TCP)
    - Sender does not send until receipt is acknowledged
    - Amount sent is based on receiver's current available buffer size – so receiver won't be flooded.
  - Be strict when sending and tolerant when receiving

WEB

- Goal is Universal information retrieval
- Decentralized
- Modular
- Extensible
- Scalable
- Accessible
- Forward/backwards compatibility

Defining The World Wide Web

- A wide-area hypertext, multimedia information retrieval system that provides access to a large universe of documents
- A uniform way of accessing and viewing some information on the Internet
- The WWW
  - creates a world in which information has a reference by which it can be accessed

Web and HTTP History-Incomplete

- WWW: Allows computer users to locate and view multimedia-based documents
  - Introduced in 1990 by *Tim Berners-Lee for astrophysicist to share documents*
- 1990 – World Wide Web project
  - Tim Berners-Lee starts project at CERN
  - Demonstrates browser/editor accessing hypertext files
  - HTTP 0.9 defined, supports only hypertext, linked to port 80 (no images)
- 1991 – first web server outside Europe
  - CERN releases WWW, installed at SLAC
- 1992 – HTTP 1.0, supports images, scripts as well
- 1993 – Growth phase
- 1994 – CERN and MIT agree to set up WWW Consortium
- 1999 – HTTP 1.1, supports open ended extensions

History of the Internet

- *ARPAnet*
  - Implemented in late 1960's by ARPA (Advanced Research Projects Agency of DOD)
  - Networked computer systems of a dozen universities and institutions with 56KB communications lines
  - Grandparent of today's Internet
  - Intended to allow computers to be shared
  - Became clear that key benefit was allowing fast communication between researchers – *electronic-mail* (*email*)
- ARPA's goals
  - Allow multiple users to send and receive info at same time
  - Network operated *packet switching* technique
    - Digital data sent in small packages called *packets*
    - Packets contained data, address info, error-control info and sequencing info
    - Greatly reduced transmission costs of dedicated communications lines

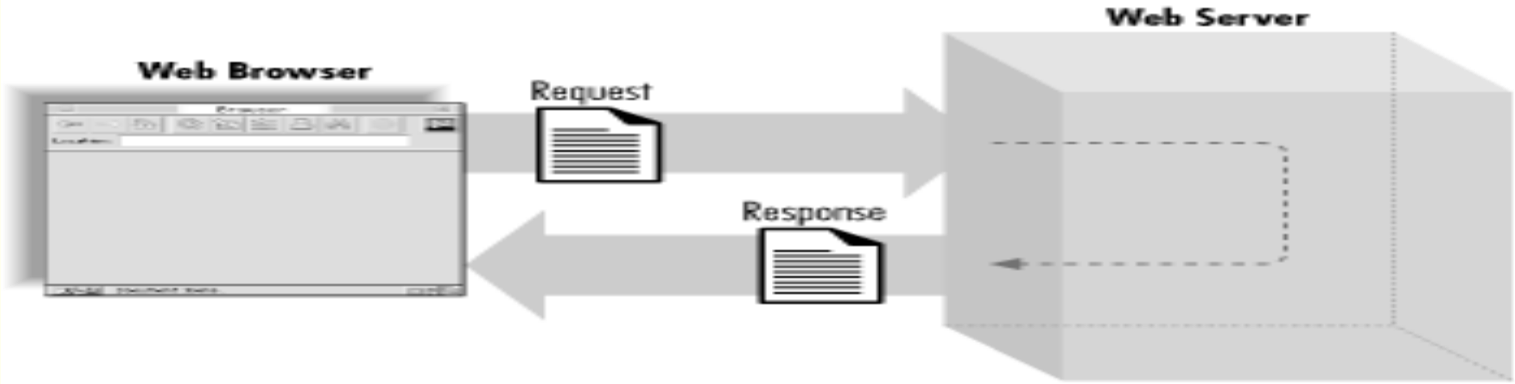
History of the Internet (II)

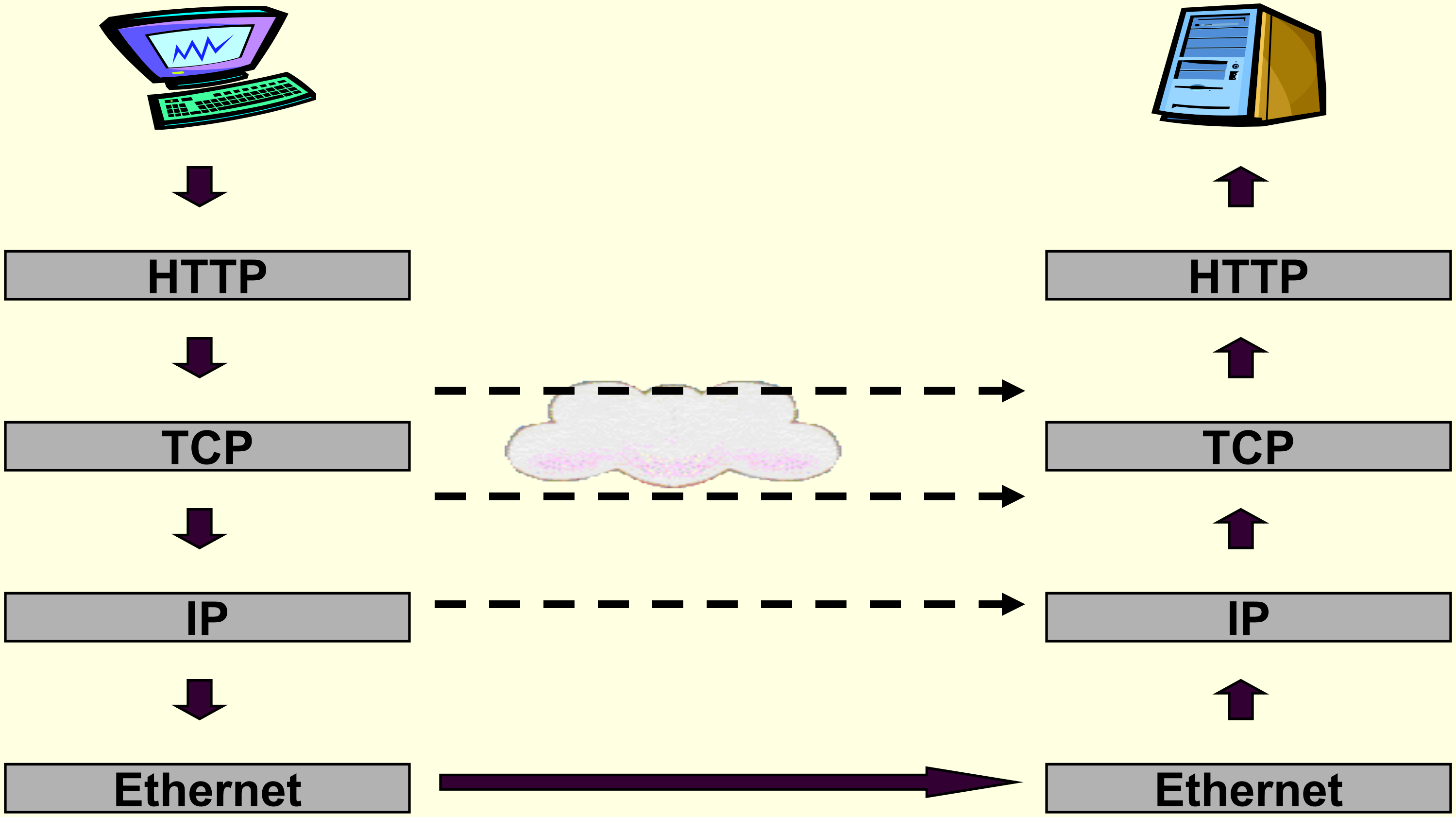
  - Network designed to be operated without centralized control
    - If portion of network fails, remaining portions still able to route packets
- ***T**ransmission **C**ontrol **P**rotocol* (TCP)
  - Name of protocols for communicating over ARPAnet
  - Ensured that messages were properly routed and that they arrived intact
- Organizations implemented own networks
  - Used both for intra-organization and communication
- Huge variety of networking hardware and software appeared
  - ARPA achieved inter-communication between all platforms with development of the *IP*
    - *Internetworking Protocol*
    - Current architecture of Internet
  - Combined set of protocols called *TCP/IP*

Major Web Technology Components

- Client/server architecture
  - where client programs interact with web servers
- Network protocol
  - HTTP, Hypertext Transfer Protocol, is the language understood by browsers and web servers
  - designed to move quickly from document to document
- Addressing system (Uniform Resource Locators)
  - http://domain/directory/file.html
- Markup Language
  - every web server understands and every browser displays
  - includes support for Hypertext and multimedia

Network Protocol (Http)





HTTP

TCP

IP

Ethernet

HTTP

TCP

IP

Ethernet

The WWW Server

- Web browsers and Web servers communicate according to a protocol known as HTTP (HyperText Transfer Protocol)
  - The current HTTP protocol is version 2
- The Web server is a software system running on a machine often called the Web server, don’t confuse them (e.g IIS)
- A web server can
  - receive and reply to HTTP requests
  - retrieve documents from specified directories
  - run programs in specified directories
  - handle limited forms of security
- A web server does not
  - know about the contents of a document, links in a document, images in a document or whether a particular file, e.g. a *.gif file, is in the correct format

Web Browsers (Agents)

- Software programs that access web's rich content
- Popular web browsers
  - Microsoft's *Internet Explorer*
  - Mozilla's *Firefox*
  - Apple's *Safari*
  - Opera Software's *Opera*
  - *Google Chrome*
- When a user submits a browser request to a web server, it sends two categories of data:
  - Data: file name of the html file or data that the user explicitly typed into an HTML form.
    - For example: registration information.
  - HTTP Request Header Data: Data that is automatically appended to the HTTP Request from the client.
    - For example: cookies, browser type, etc,

Web and HTTP

- Web page consists of objects

- Object can be HTML file, JPEG image, Java applet, audio file,…

- Web page consists of base HTML-file which includes several referenced objects

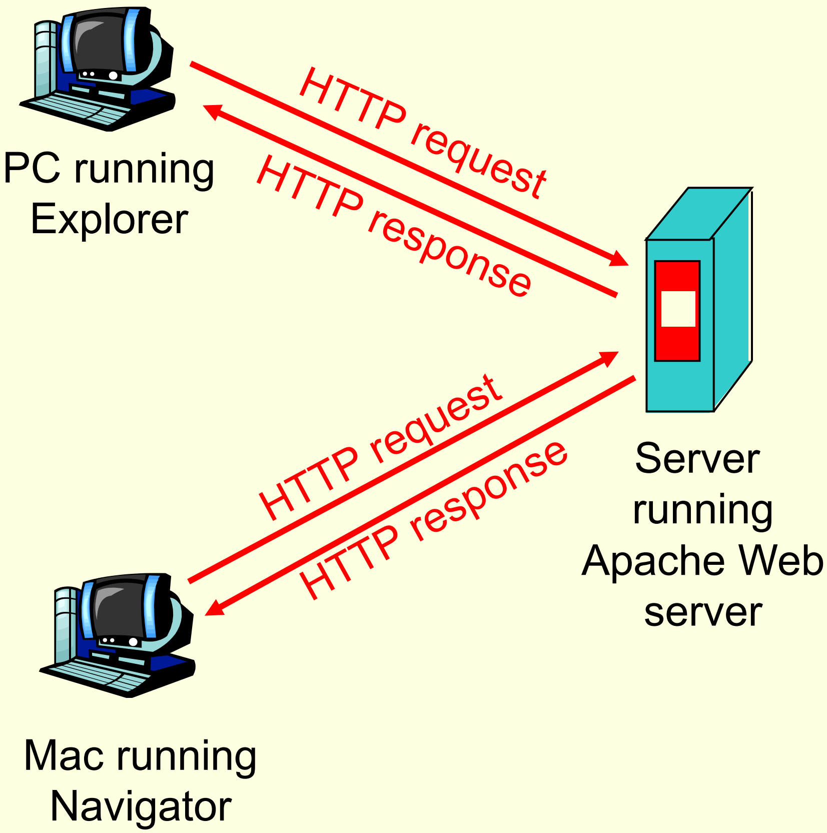
- Each object is addressable by a URL
  - Example URL:

$$\underbrace{\texttt{www.someschool.edu}}_{\text{host name}}/\underbrace{\texttt{someDept/pic.gif}}_{\text{path name}}$$

HTTP overview

HTTP: hypertext transfer protocol

- Web’s application layer protocol
- simple structure:
  - client sends a request
  - server returns a reply.
  - support multiple request-reply exchanges over a single TCP connection
- client/server model
  - *client:* browser that requests, receives, “displays” Web objects
  - *server:* Web server sends objects in response to requests



http://web-sniffer.net/

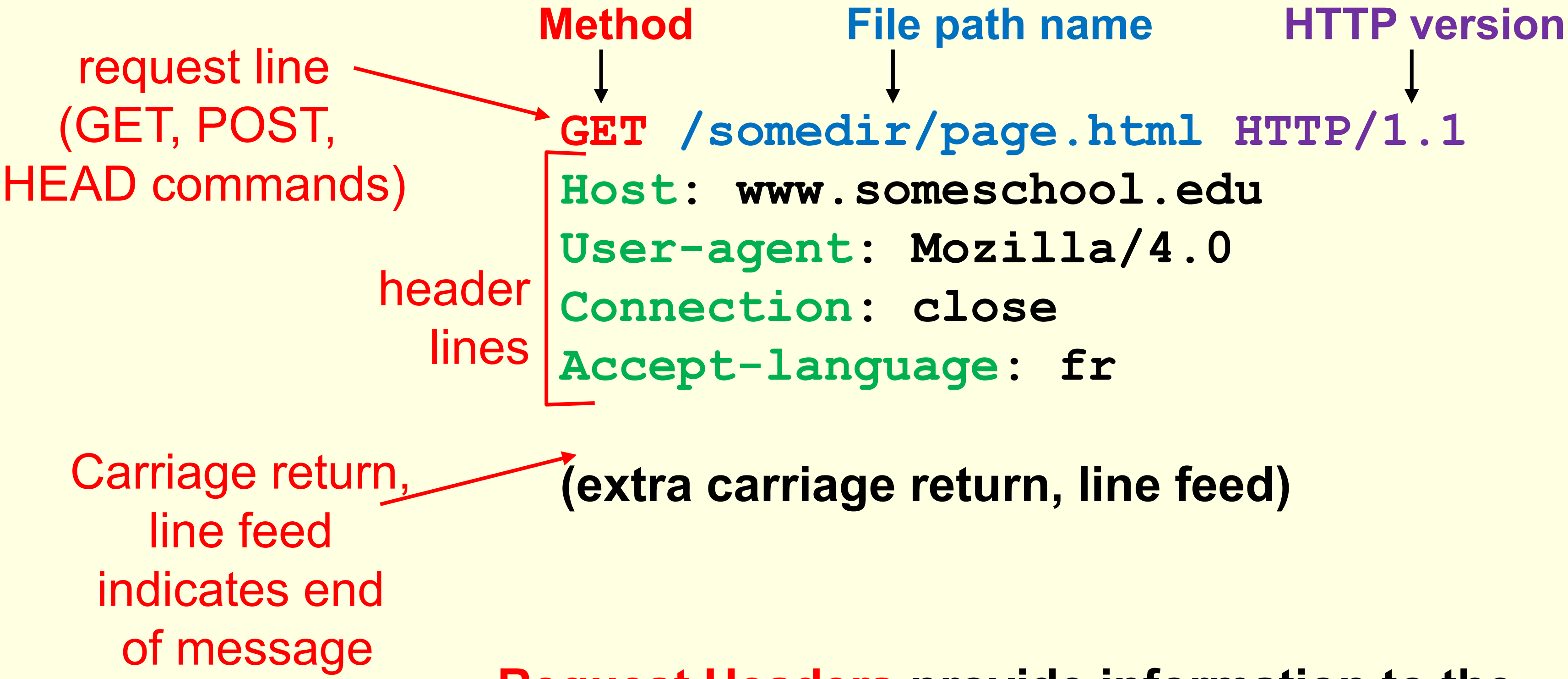
[See wikipiedia]() for more details

Typical HTTP Transaction

- Client browser finds a machine address from an internet Domain Name Server (DNS).
- Client and Server open TCP/IP socket connection. Server waits for a request.
- Browser sends a **verb** and an **object (also headers)**:
  - GET XYZ.HTM or POST form.html
  - If there is an error, server can send back an HTML-based explanation.
  - If there is no error, Server applies **headers** to a returned HTML file and delivers to browser.
- Accept: image/gif, ……, application/msword, */*
- Accept-Language: ar-sa
- Referer: www.me.com……
- Accept-Encoding: gzip, deflate

HTTP request message

- two types of HTTP messages: *request, response*
- HTTP request message:
  - ASCII (human-readable format)



Method

File path name

HTTP version

request line (GET, POST, HEAD commands)

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: fr
```

header lines

(extra carriage return, line feed)

Carriage return, line feed indicates end of message

Request Headers provide information to the server about the client

what kind of client

what kind of content will be accepted

who is making the request

Uploading form input

Post method:

- Web page often includes form input
- Input is uploaded to server in entity body

URL method:

- Uses GET method
- Input is uploaded in URL field of request line:

```
GET /path/script.cgi?home=Cosby&favorite+flavor=flies HTTP/1.0
User-Agent: my_soft/1.0
[blank line here]
```

```
POST /path/script.cgi HTTP/1.0
User-Agent: my_soft/1.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 32

home=Cosby&favorite+flavor=flies
```

A Typical HTTP Transaction (cont.)

- you will see what I received from my http://java.sun.com/

Status line:

HTTP/1.1 200 OK

Response headers:

Content-Type: text/html

Server: Oracle-Application-Server-11g Oracle-Web-Cache-11g/11.1.1.2.0 (TH;max-age=300+0;age=157;ecid=830374227676171540)

Date: Fri, 01 Oct 2016 11:46:59 GMT

Transfer-Encoding: chunked

Connection: keep-alive

Transfer-Encoding: chunked

- you will see what I received from my https://java.sun.com/x

Status line:

HTTP/1.1 404 Not found

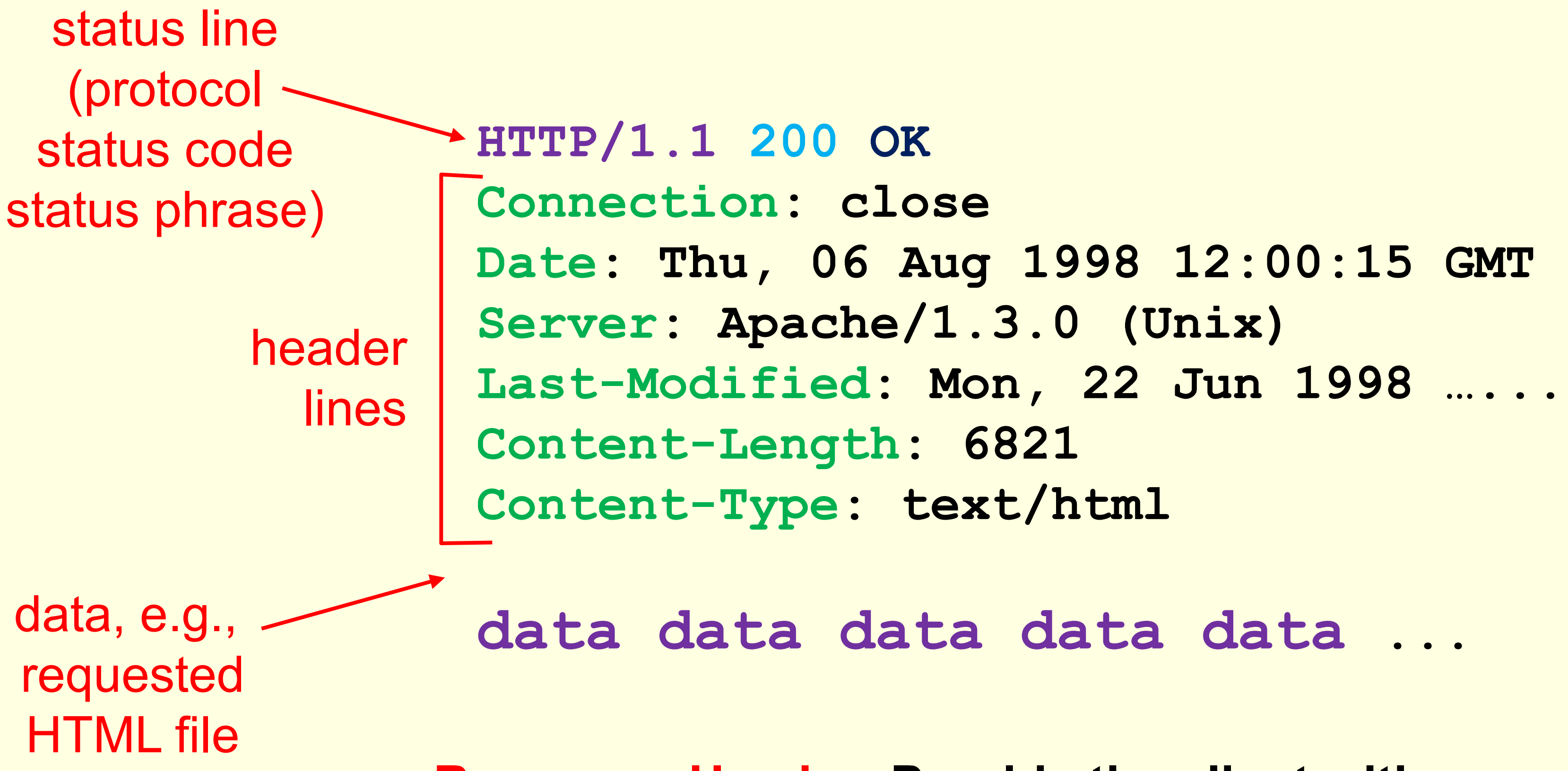
Response headers:

Server: Sun-Java-System-Web-Server/7.0

Date: Fri, 01 Oct 2016 11:51:55 GMT

Transfer-encoding: chunked

HTTP response message



status line (protocol status code status phrase)

header lines

data, e.g., requested HTML file

```
HTTP/1.1 200 OK
Connection: close
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 22 Jun 1998 …...
Content-Length: 6821
Content-Type: text/html

data data data data data ...
```

Response Header: Provide the client with information about the returned *entity* (document).

what kind of document
how big the document is
how the document is encoded
when the document was last modified

HTTP response status codes

- A few sample codes:

- **100 Continue but**
  - the client should respond with some other action

200 OK

- request succeeded, requested object later in this message

301 Moved Permanently

- requested object moved, new location specified later in this message (Location:)

400 Bad Request

- request message not understood by server

404 Not Found

- requested document not found on this server

505 HTTP Version Not Supported