Web Technology & Programming

Introduction

LECTURER: **KRUNAL PATEL**

ICCT ENGG. COLLEGE

NEW VALLABH VIDYANAGAR

Introduction

First Use of Internet

- October 1969: First packets on the ARPANet from UCLA to Stanford. Starts to send "LOGIN", but it crashes on the G.
- Packet Switching: Leonard Kleinrock (UCLA) thinks he did, Donald Davies and Paul Baran, Edelcrantz's signalling network (1809) sort of did it
- Internet Protocol: Vint Cerf, Bob Kahn
- Vision, Funding: J.C.R. Licklider, Bob Taylor
- Government: **Al Gore** (first politician to promote Internet, 1986; act to connect government networks to form "Interagency Network")

Internet Basic

- ***The Internet consist of two types of computers Servers and Clients***
  - Computers which offer information to be read are called servers
  - Computers that read the information offered are called clients

- ***Servers run special software called (Web Server) that allows to***
  - Respond to client requests for information
  - Accept data from clients

  Ex.

  IIS

  Apache Web Server

Internet Basic

- ***Clients run special software(Browser Software) that allows them to***
  - Locate the appropriate server
  - Query the server for the information to be read

Ex.

Netscape

IE

Crome

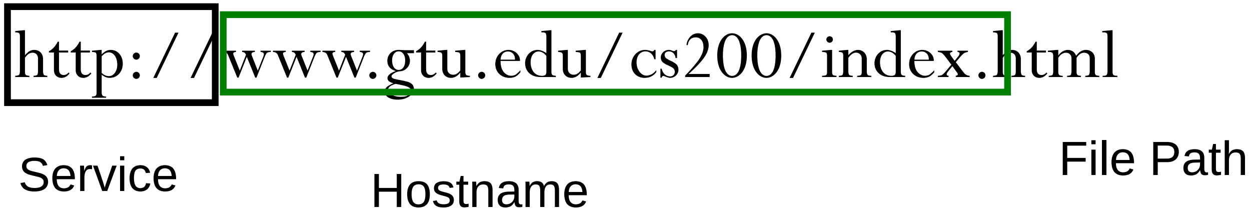
Mozzila

The World Wide Web

- Tim Berners-Lee, CERN (Switzerland)
- First web server and client, 1990
- Established a ***common language*** for sharing information on computers
- Lots of previous attempts (Gopher, WAIS, Archie, Xanadu, etc.)
- The WWW today is a distributed client/server service, in which a client using a browser can access a service using a server. However, the service provided is distributed over many locations called sites.

The World Wide Web

- World Wide Web succeeded because it was **simple**!
  - Didn't attempt to maintain links, just a common way to name things
  - Uniform Resource Locators (URL)



http://www.gtu.edu/cs200/index.html

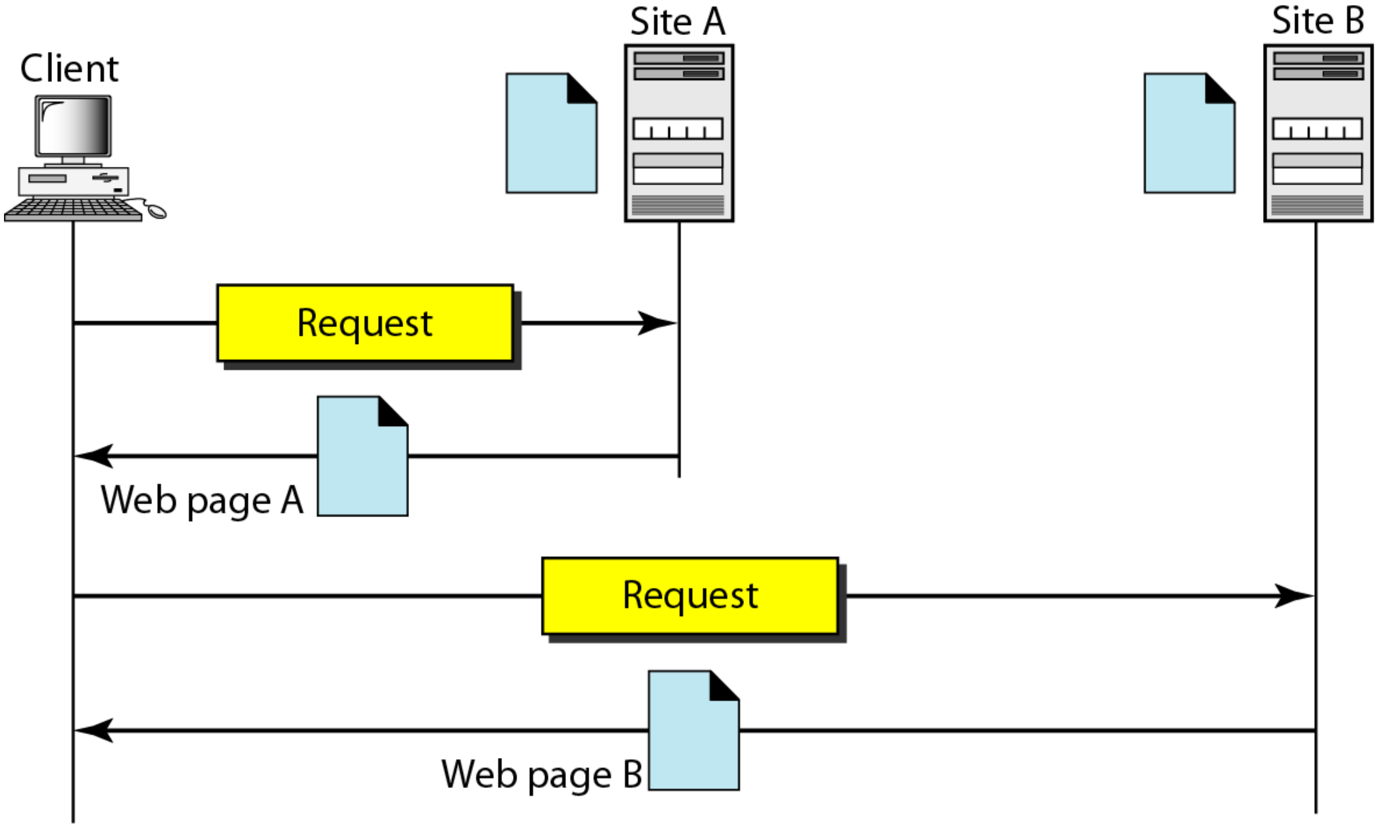
Service

Hostname

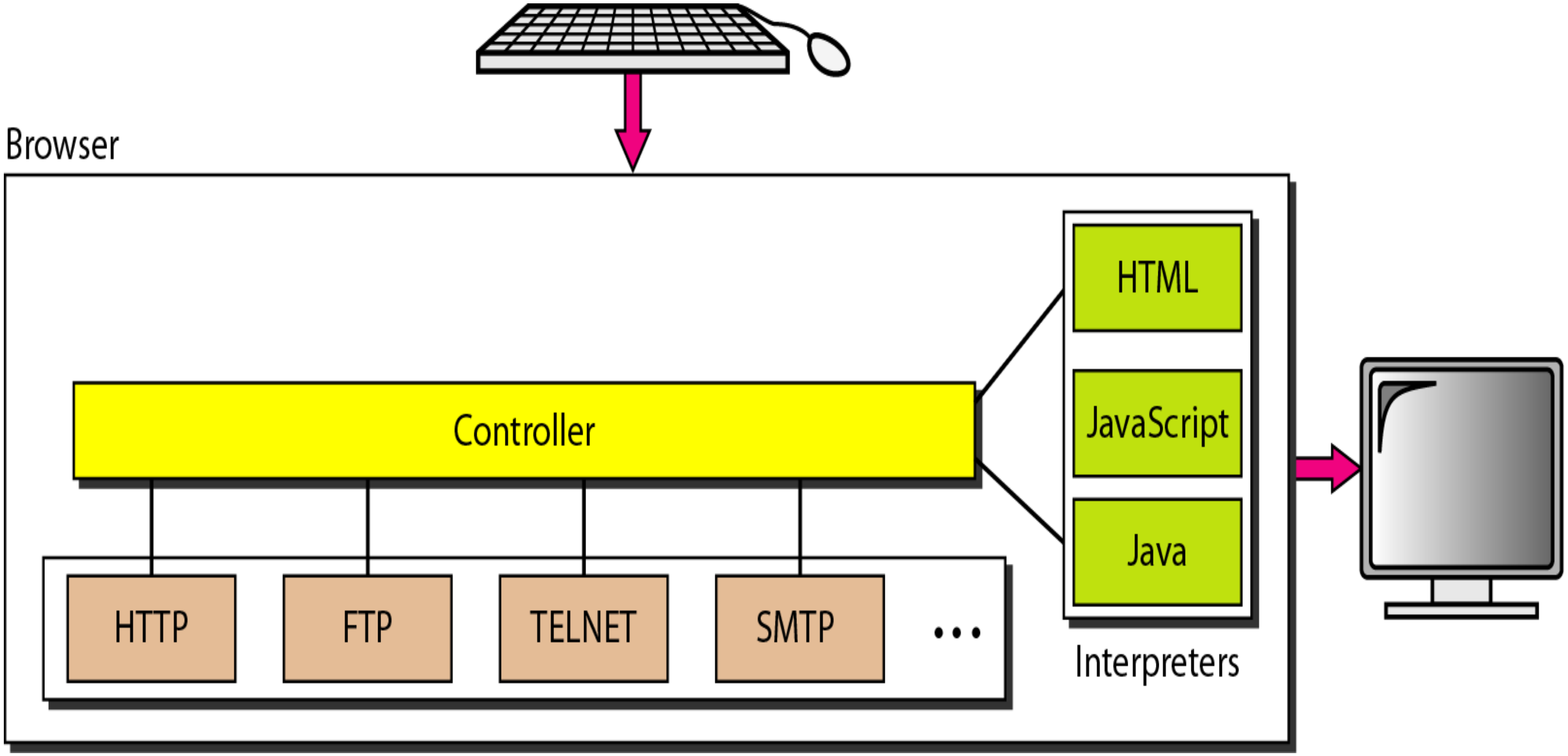
File Path

HyperText Transfer Protocol

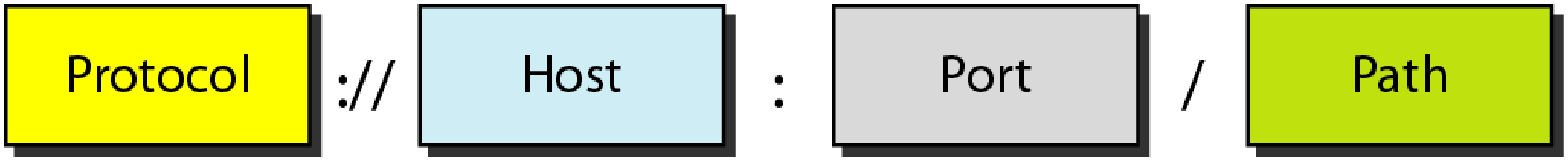
Architecture of www



Browser


Browser
Controller
HTTP
FTP
TELNET
SMTP
...
HTML
JavaScript
Java
Interpreters

URL

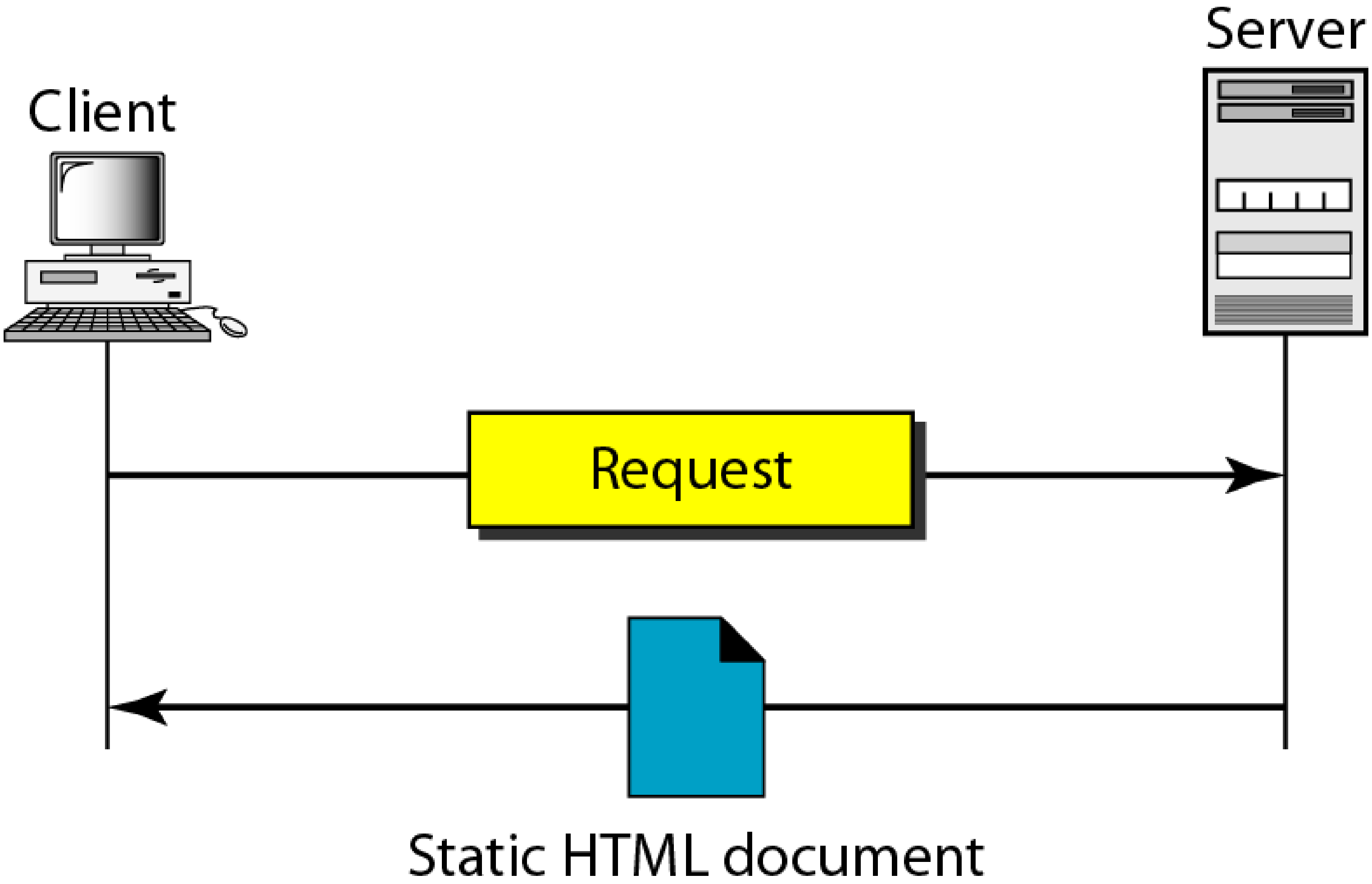


Protocol :// Host : Port / Path

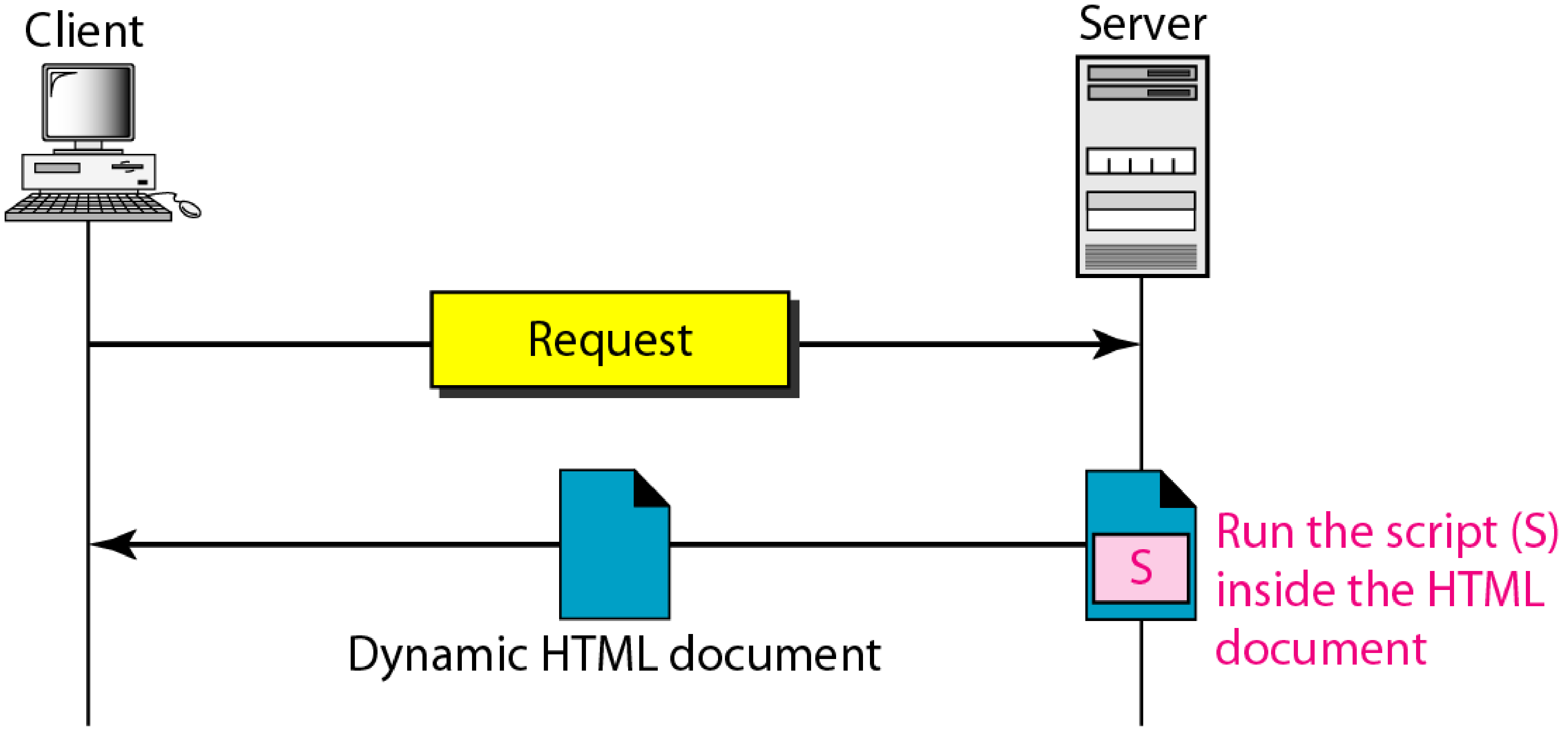
Web Documents

- The documents in the WWW can be grouped into three broad categories: **static, dynamic, and active**. The category is based on the time at which the contents of the document are determined.

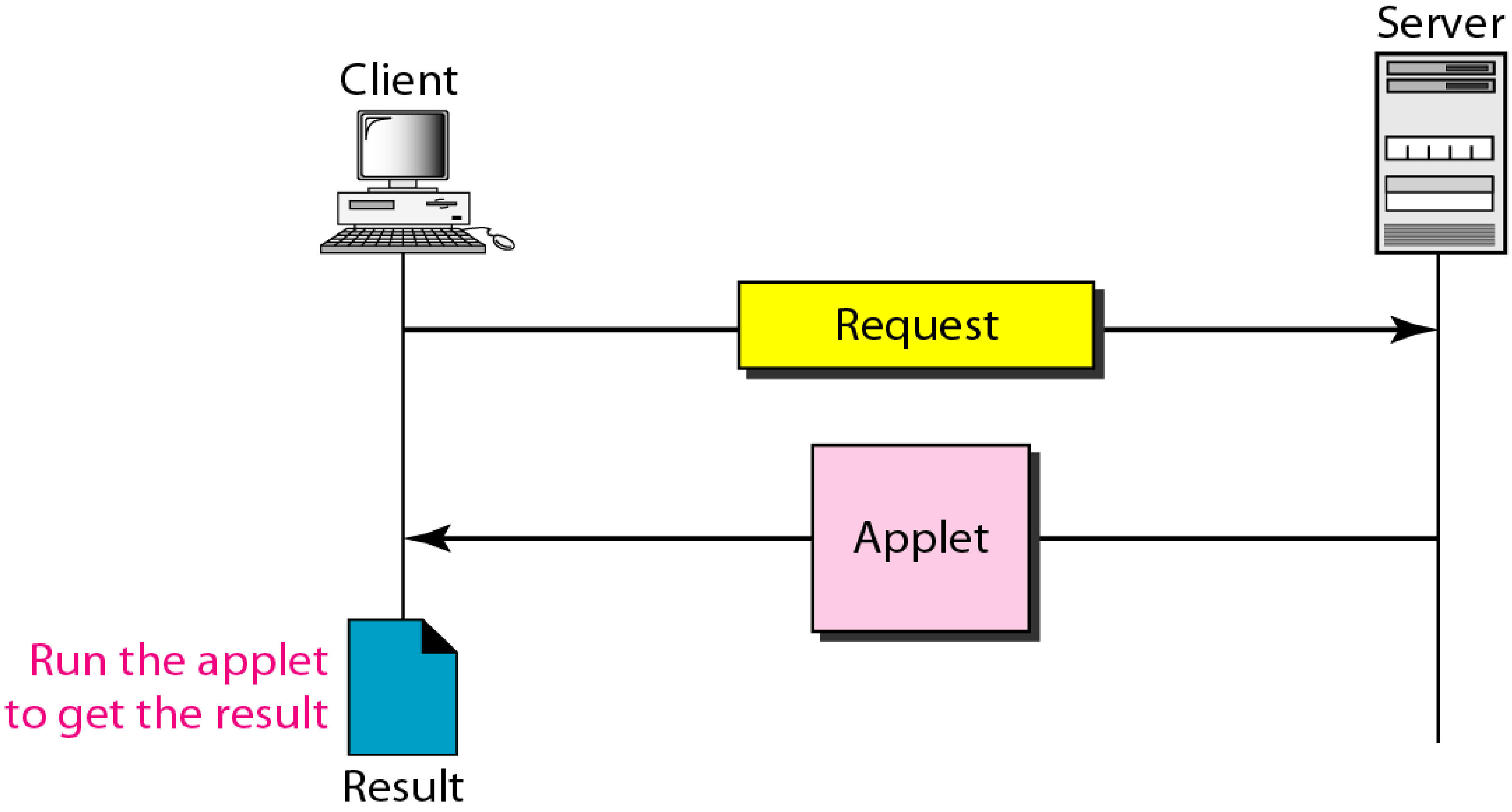
Static document



Dynamic document using CGI

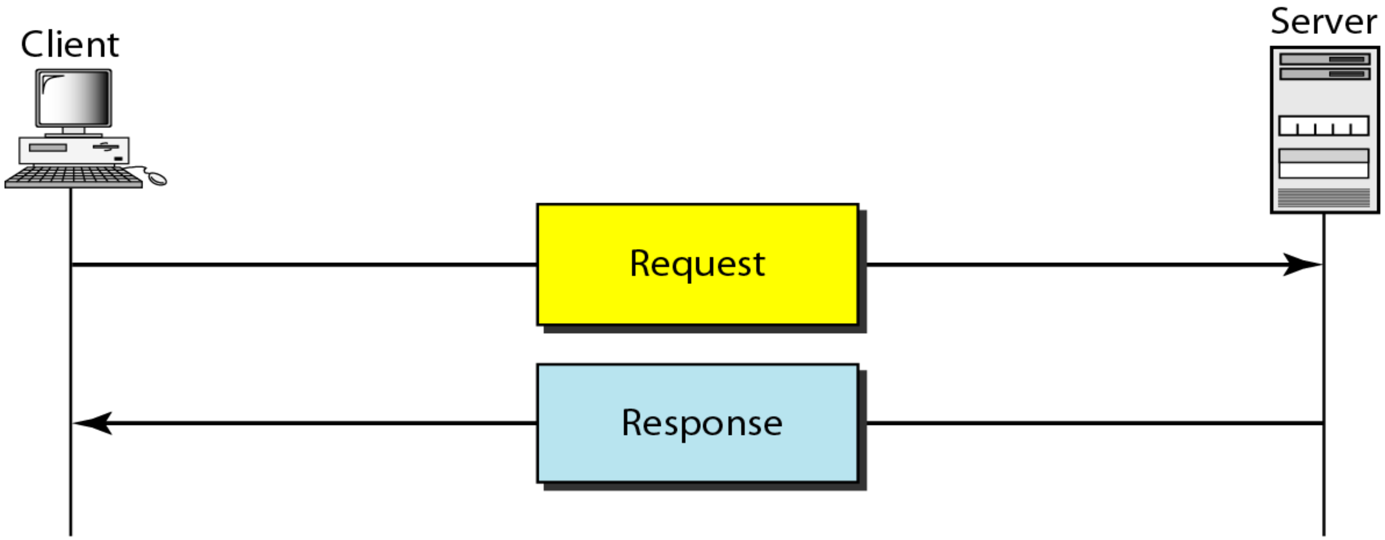


Active document using Java applet



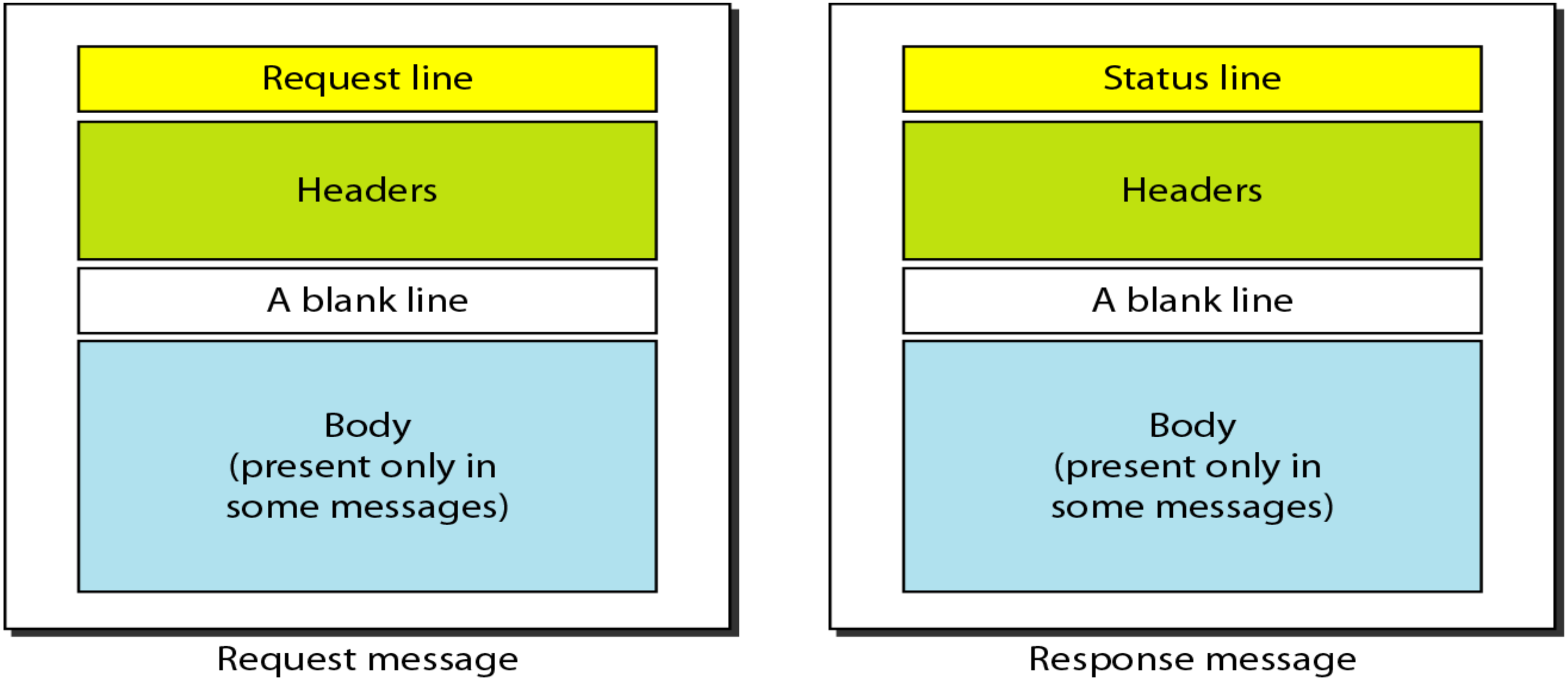
HTTP

- The Hypertext Transfer Protocol (HTTP) is a protocol used mainly to access data on the World Wide Web.
- HTTP functions as a combination of FTP and SMTP.
- HTTP uses the services of TCP on well-known port 80.
- *HTTP transaction*



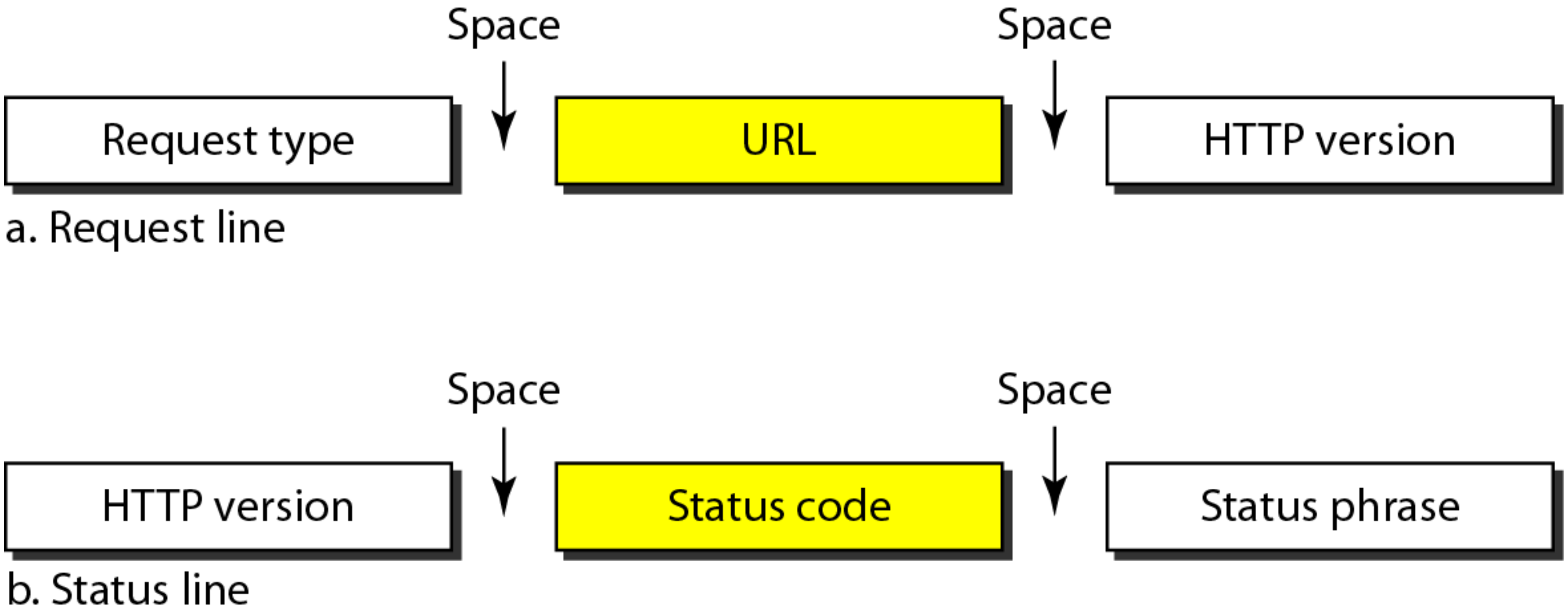
HTTP

- *Request and response messages*



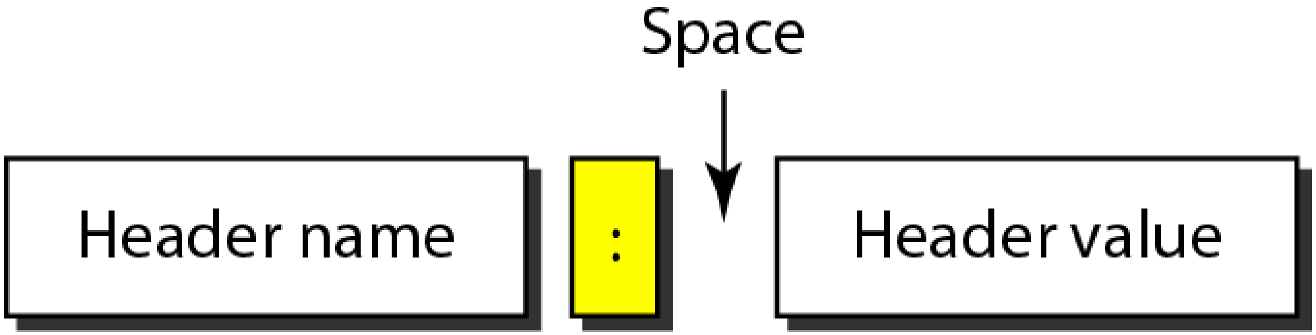
HTTP

- *Request and status lines*



HTTP

- *Header format*



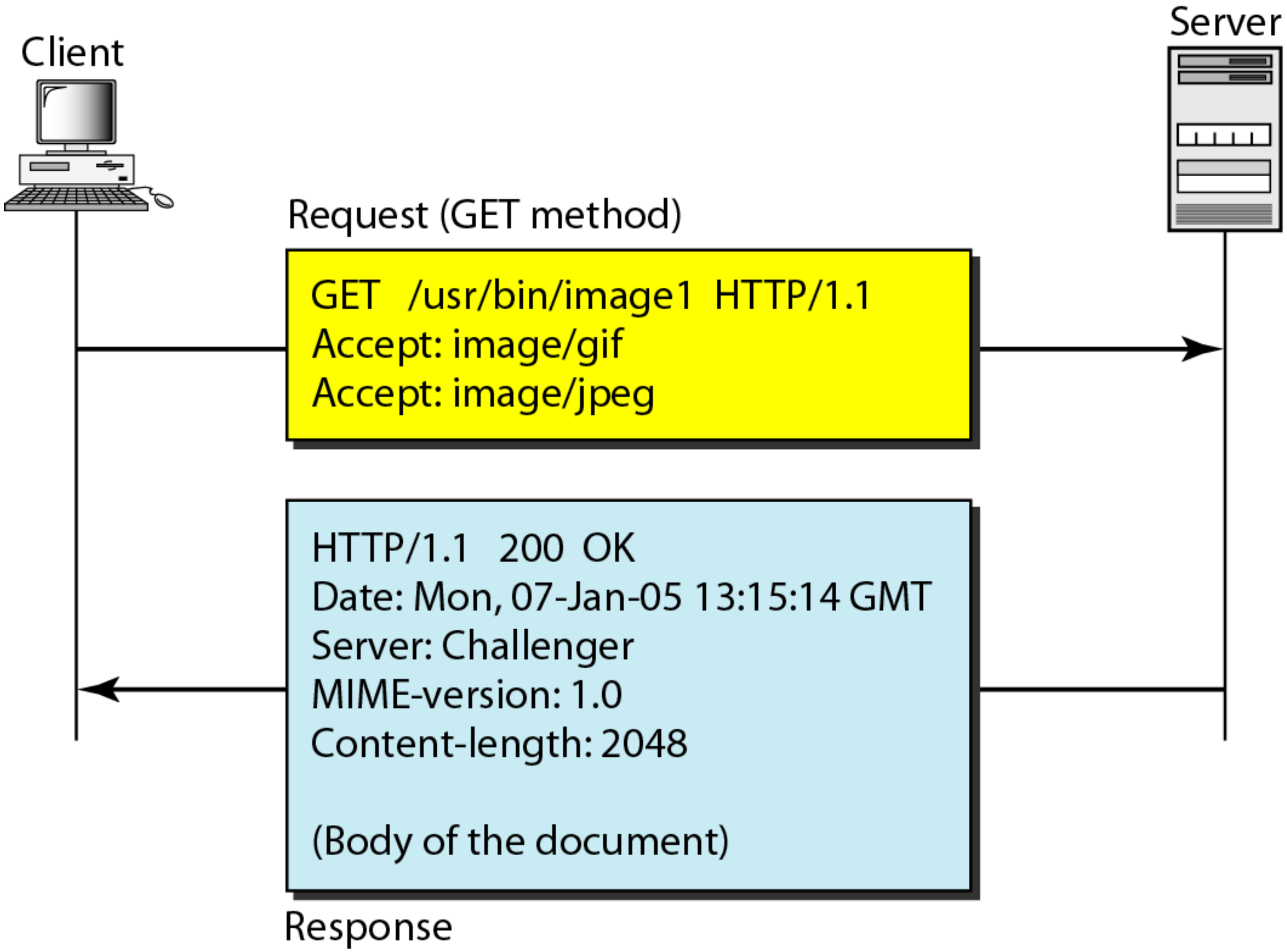
- *General headers*

Header	*Description*
Cache-control	Specifies information about caching
Connection	Shows whether the connection should be closed or not
Date	Shows the current date
MIME-version	Shows the MIME version used
Upgrade	Specifies the preferred communication protocol

HTTP

- This example retrieves a document.
  - We use the GET method to retrieve an image with the path /usr/bin/image1.
  - The request line shows the method (GET), the URL, and the HTTP version (1.1).
  - The header has two lines that show that the client can accept images in the GIF or JPEG format.
  - The request does not have a body.
  - The response message contains the status line and four lines of header.
  - The header lines define the date, server, MIME version, and length of the document.
  - The body of the document follows the header.

HTTP



Request (GET method)

GET /usr/bin/image1 HTTP/1.1
Accept: image/gif
Accept: image/jpeg

HTTP/1.1 200 OK
Date: Mon, 07-Jan-05 13:15:14 GMT
Server: Challenger
MIME-version: 1.0
Content-length: 2048

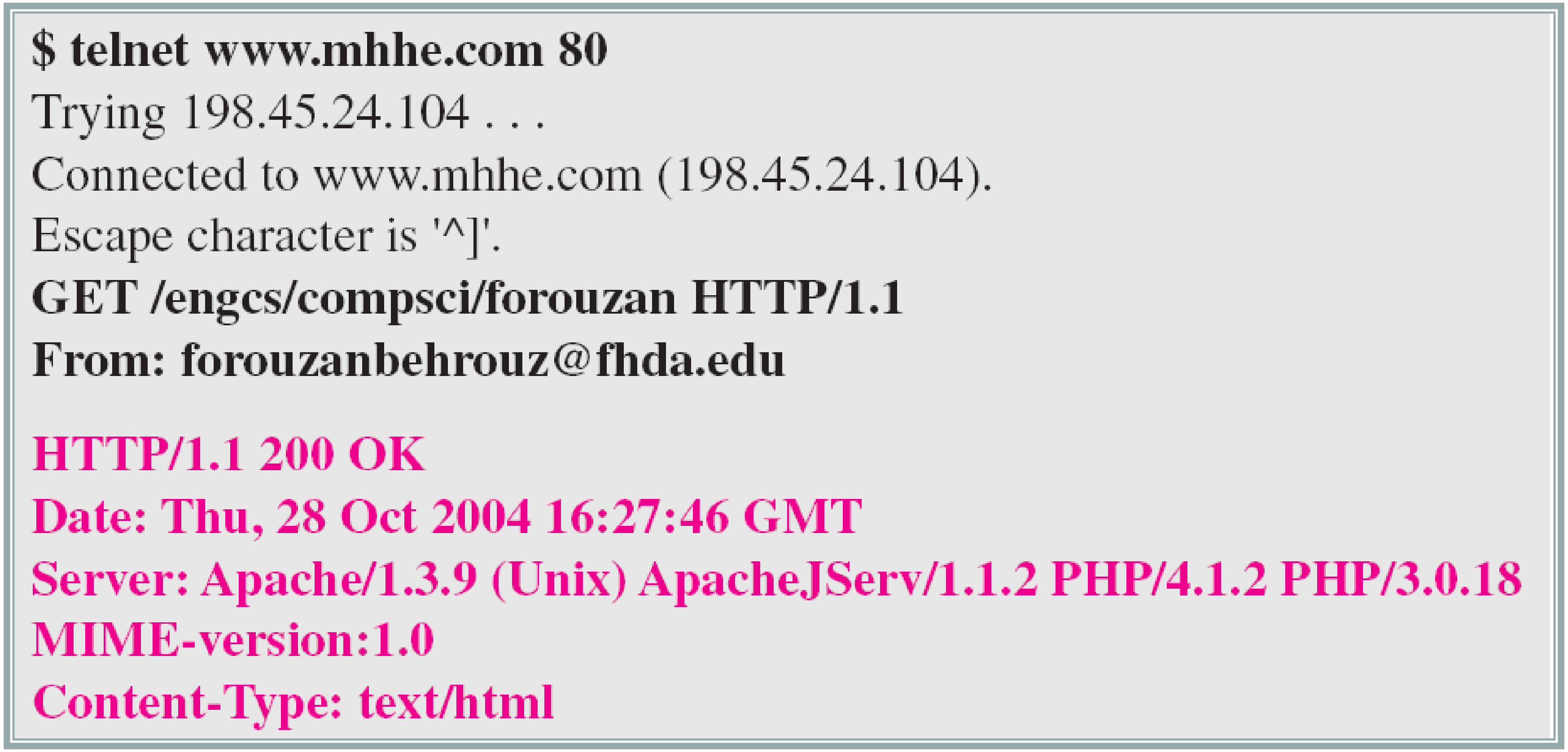
(Body of the document)

Response

HTTP

- HTTP uses ASCII characters.
  - A client can directly connect to a server using TELNET, which logs into port 80 (see next slide).
  - The next three lines show that the connection is successful.
  - We then type three lines.
    1. The first shows the request line (GET method)
    2. Second is the header (defining the host)
    3. Third is a blank, terminating the request.
  - The server response is seven lines starting with the status line.
  - The blank line at the end terminates the server response.
  - The file of 14,230 lines is received after the blank line (not shown here).
  - The last line is the output by the client.

HTTP



```
$ telnet www.mhhe.com 80
Trying 198.45.24.104 . . .
Connected to www.mhhe.com (198.45.24.104).
Escape character is '^]'.
GET /engcs/compsci/forouzan HTTP/1.1
From: forouzanbehrouz@fhda.edu

HTTP/1.1 200 OK
Date: Thu, 28 Oct 2004 16:27:46 GMT
Server: Apache/1.3.9 (Unix) ApacheJServ/1.1.2 PHP/4.1.2 PHP/3.0.18
MIME-version:1.0
Content-Type: text/html
```