# WWW and HTTP

Architecture

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.
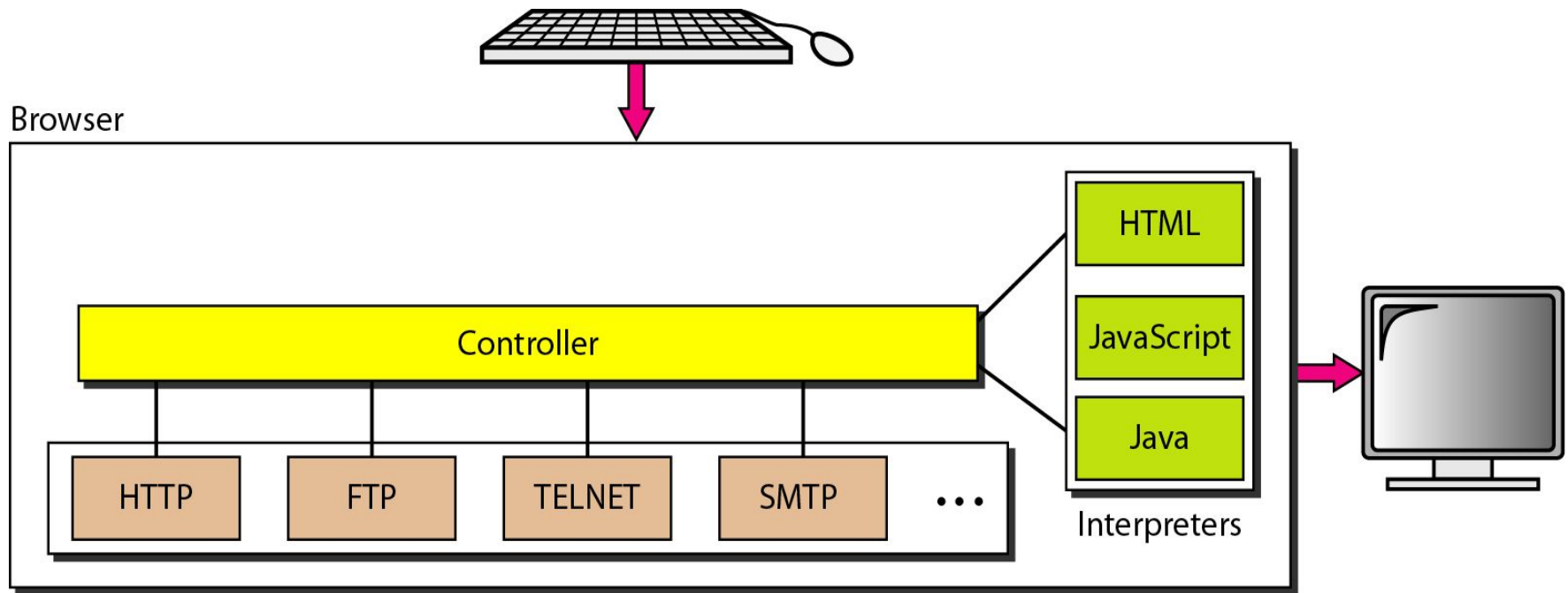
Topics discussed in this section:

- **Client (Browser)**
- **Server**
- **Uniform Resource Locator**
- **Cookies**

Architecture of WWW

Browser

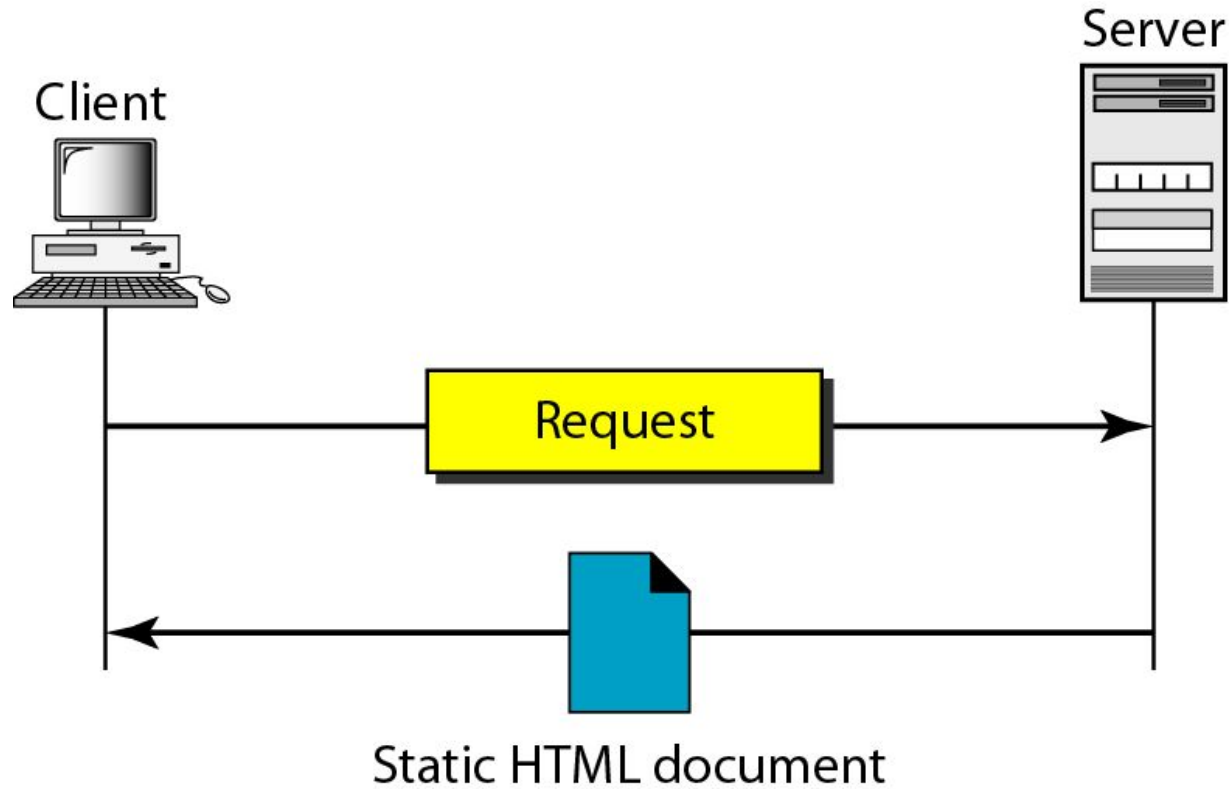# URL

Protocol :// Host : Port / Path
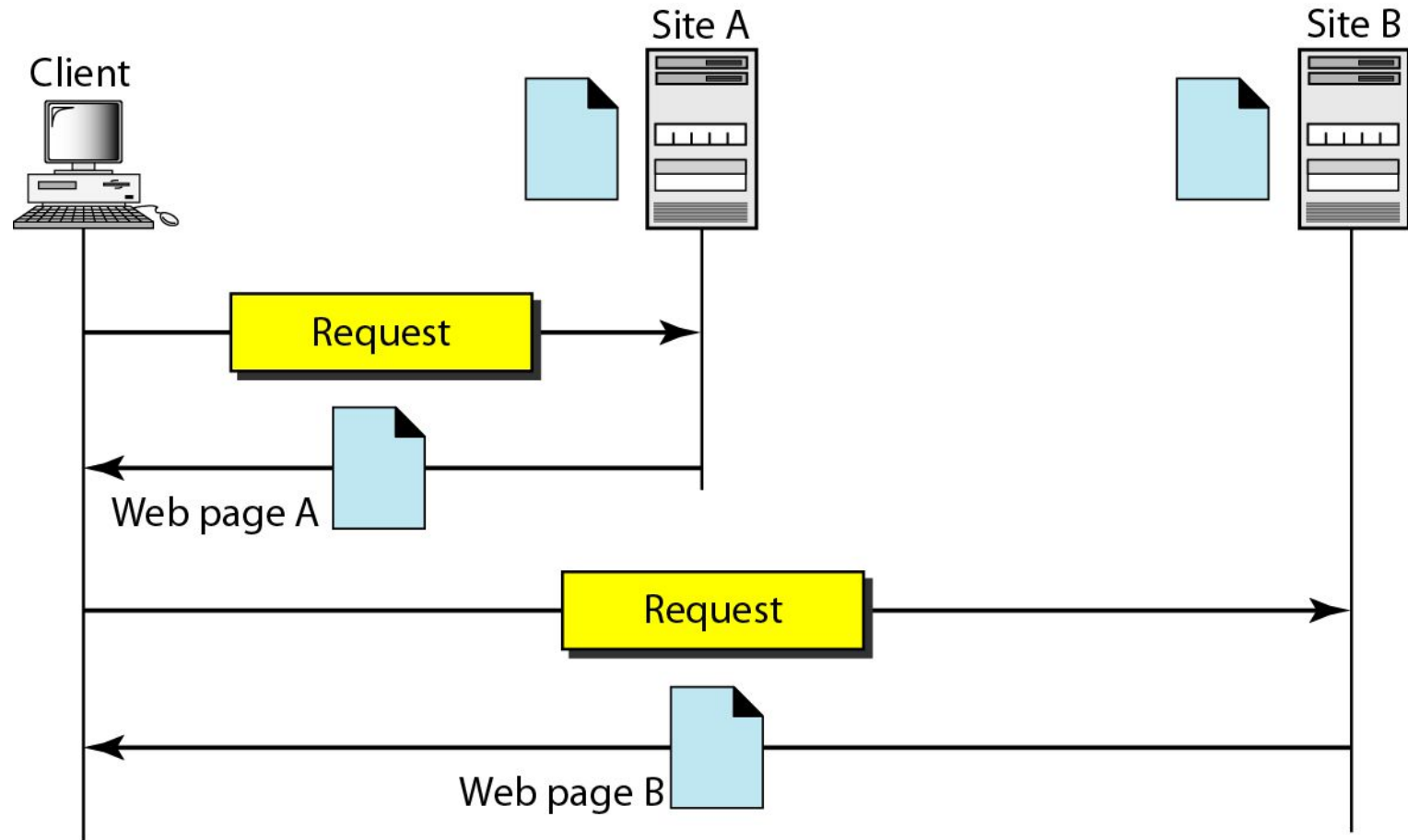
Web Documents

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.

Topics discussed in this section:

- **Static Documents**
- **Dynamic Documents**
- **Active Documents**
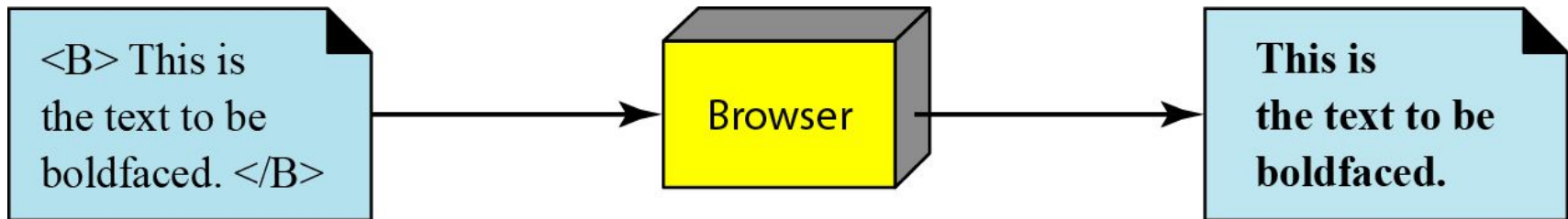
Static Document

Static HTML document

Boldface Tags

Bold tag

End bold

<B> This is the text to be boldfaced.</B>

Effect of Boldface Tags

<B> This is the text to be boldfaced. </B> → Browser → **This is the text to be boldfaced.**

Beginning and ending Tags

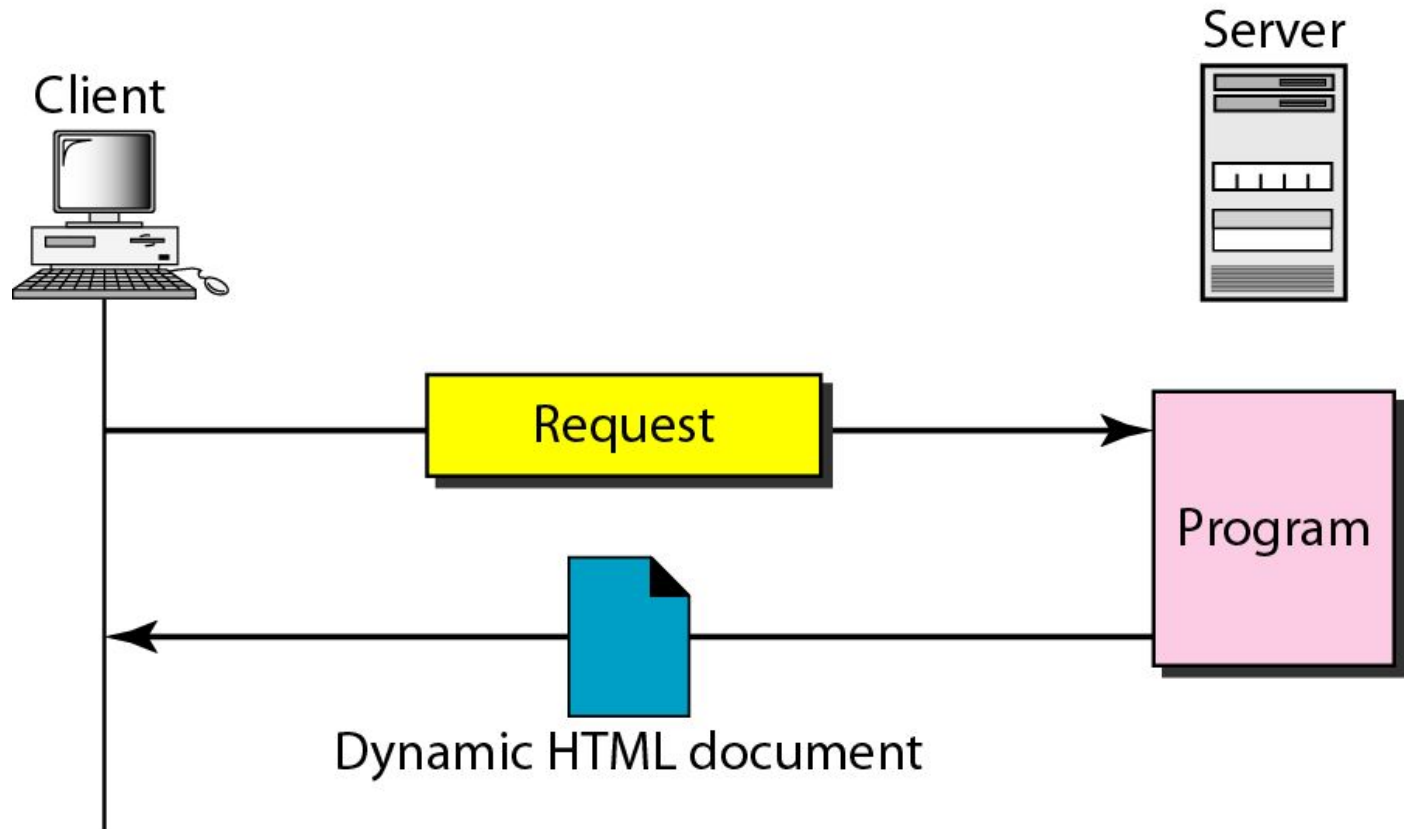
| < TagName | Attribute = Value | Attribute = Value | • • • > |

a. Beginning tag

| < /TagName > |

b. Ending tag

Dynamic document using CGI

Dynamic document using script

Client

Server

Request

Run the script (S) inside the HTML document

Dynamic HTML document
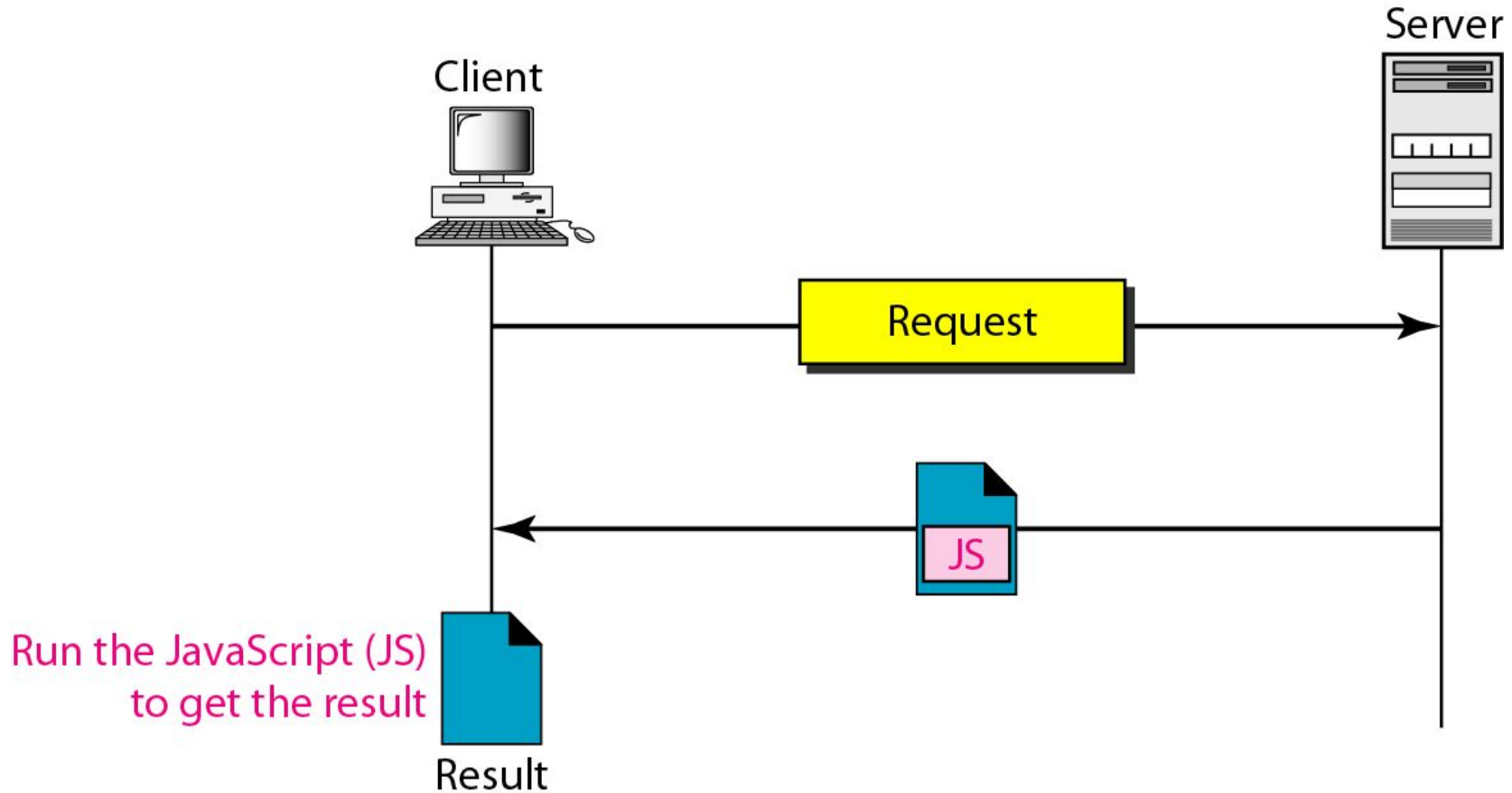
12
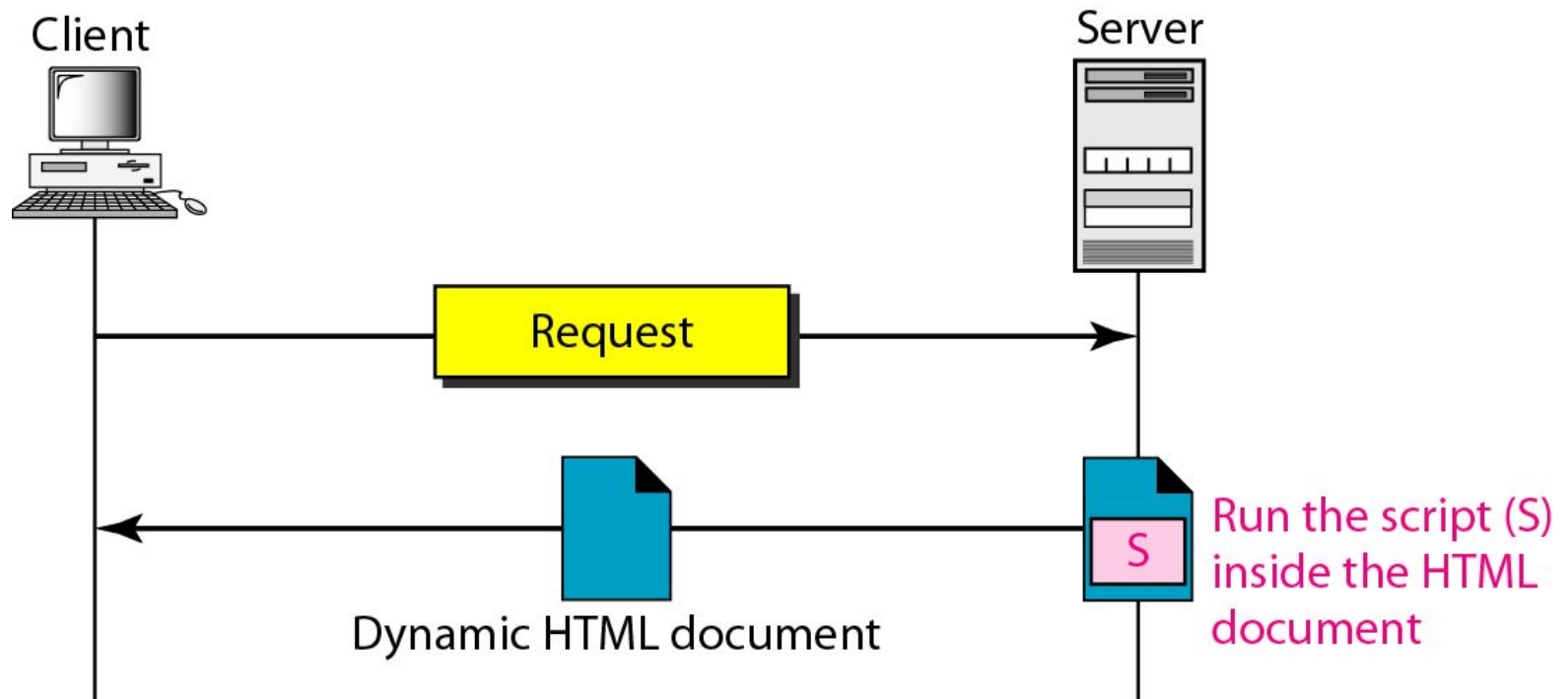
Active document using server Java applet
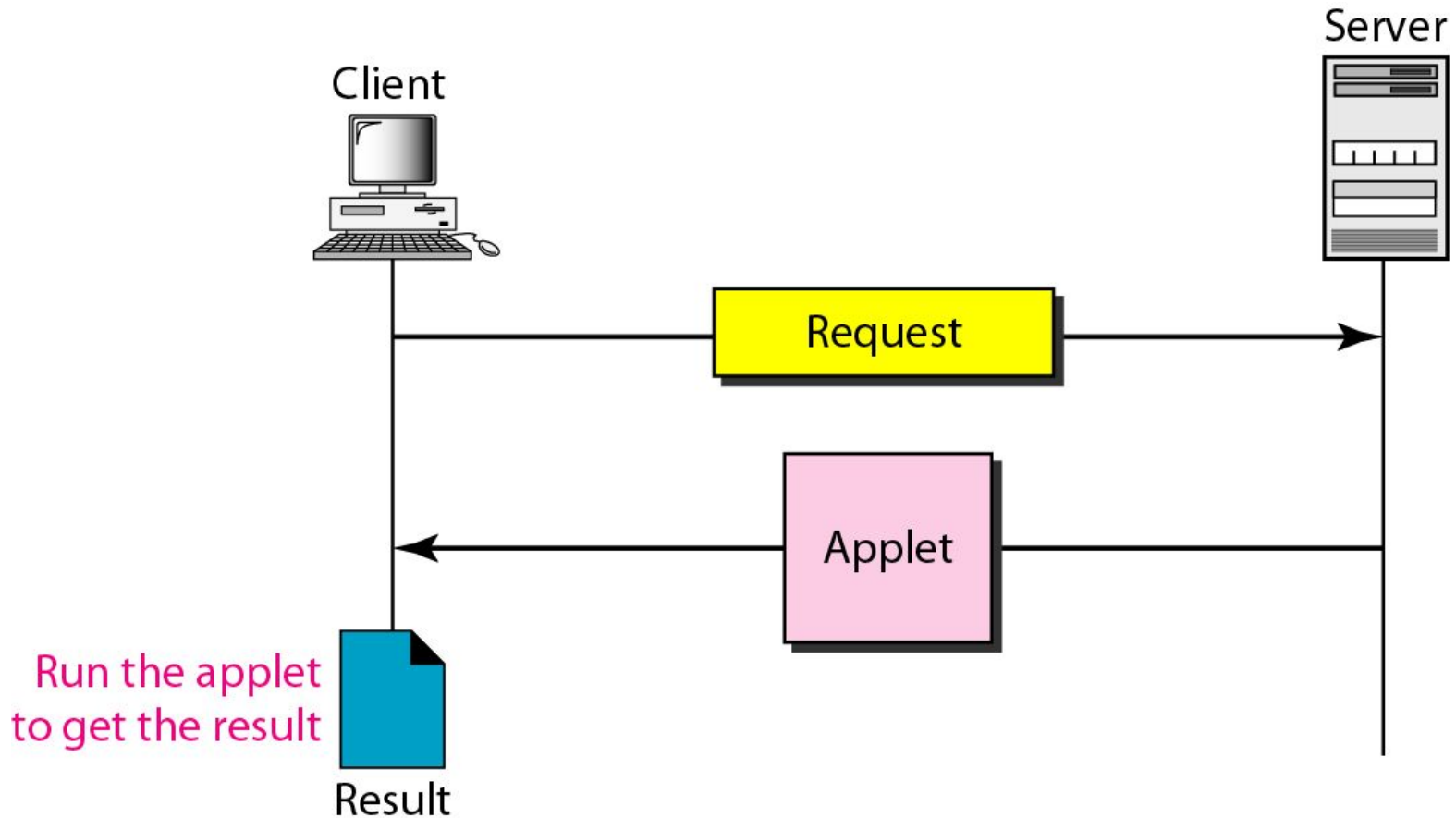
Active document using client Javascript

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.

Topics discussed in this section:

- **HTTP Transaction**
- **Persistent Versus Non persistent Connection**

HTTP transaction
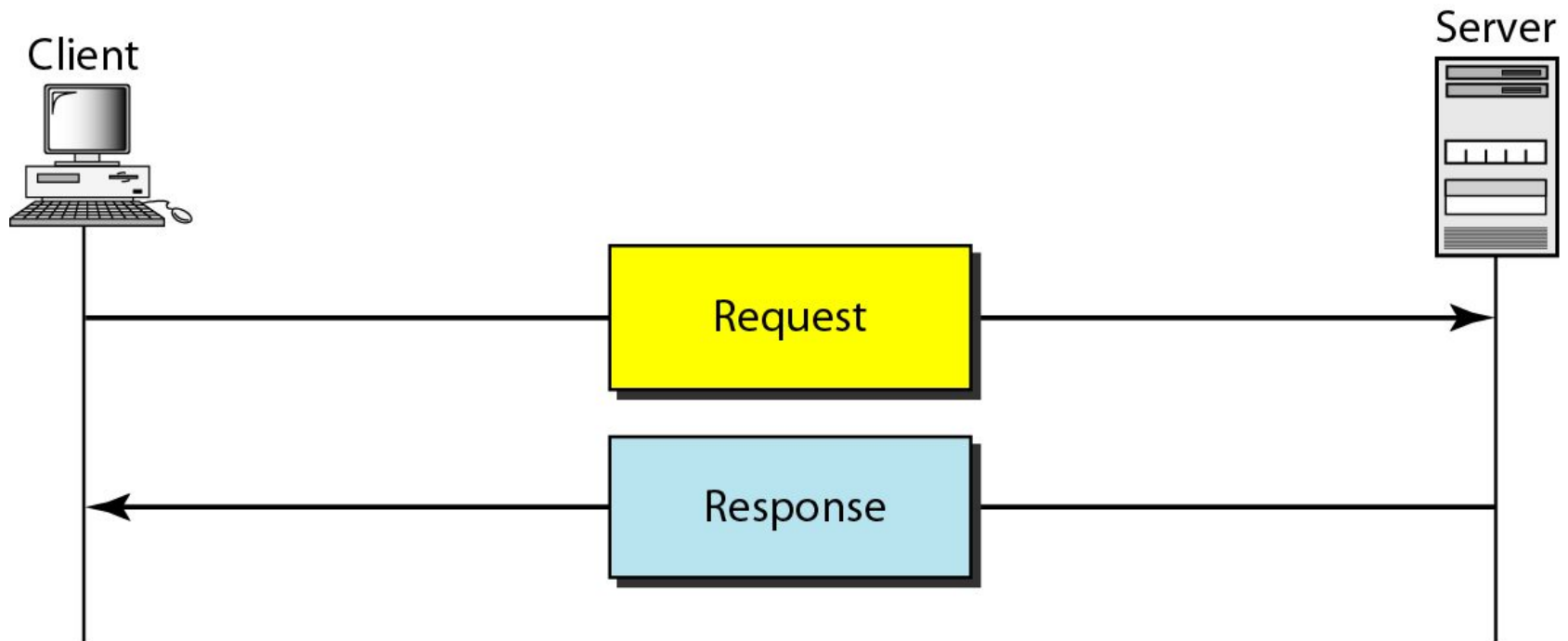
Request and response messages

Request message

Request line
Headers
A blank line
Body
(present only in
some messages)

Response message

Status line
Headers
A blank line
Body
(present only in
some messages)
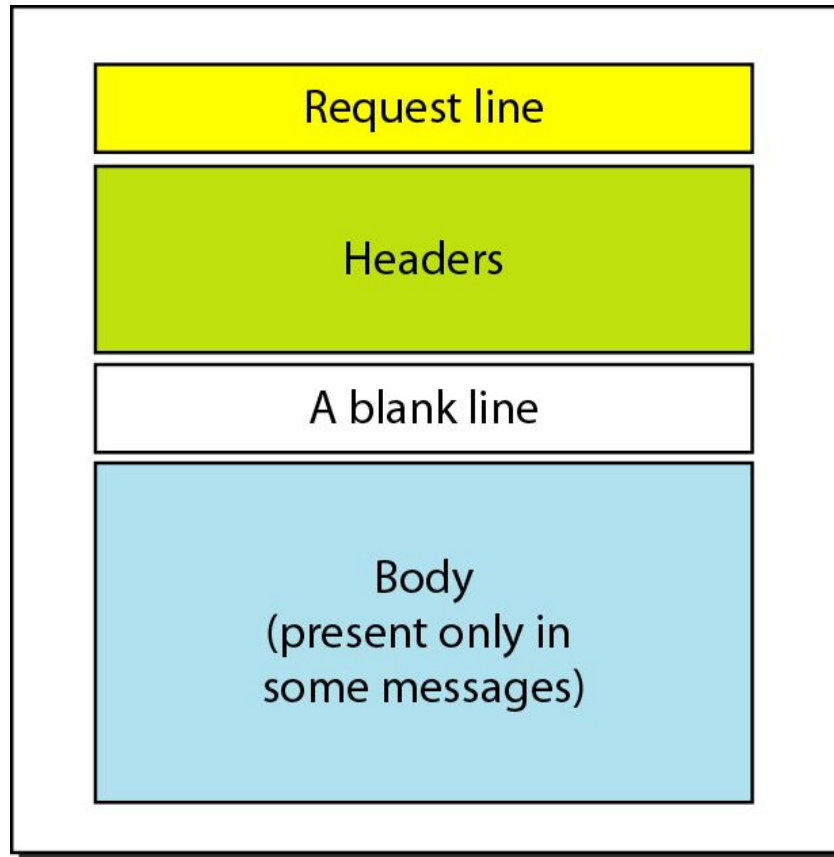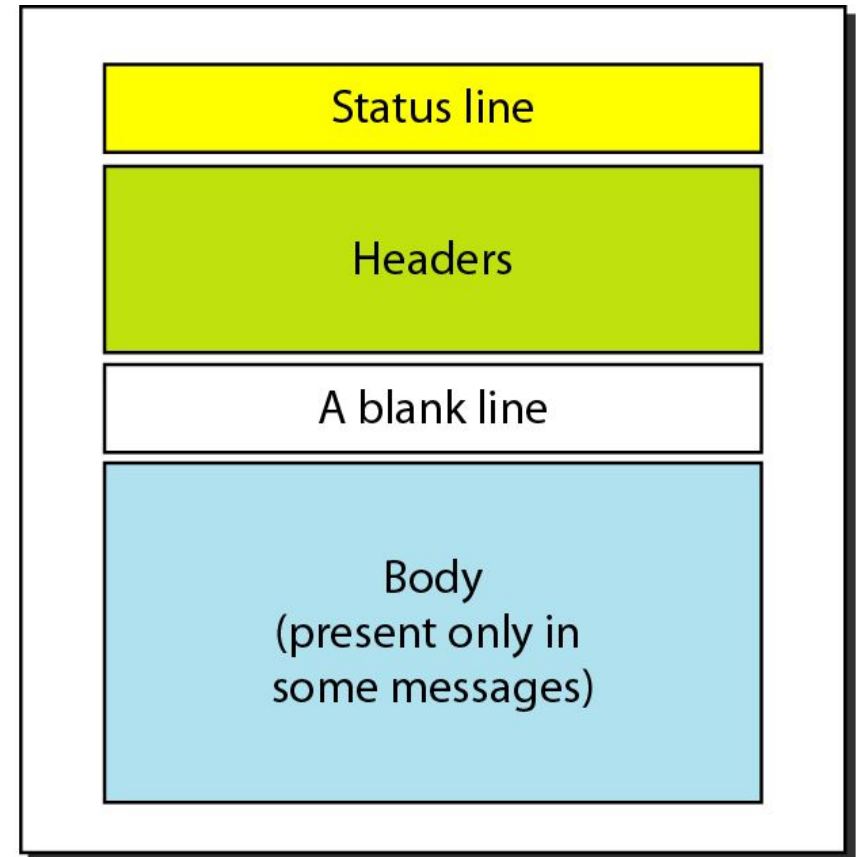
Request and status lines

Space →

| Request type | Space ↓ | URL | Space ↓ | HTTP version |

a. Request line

| HTTP version | Space ↓ | Status code | Space ↓ | Status phrase |

b. Status line

18
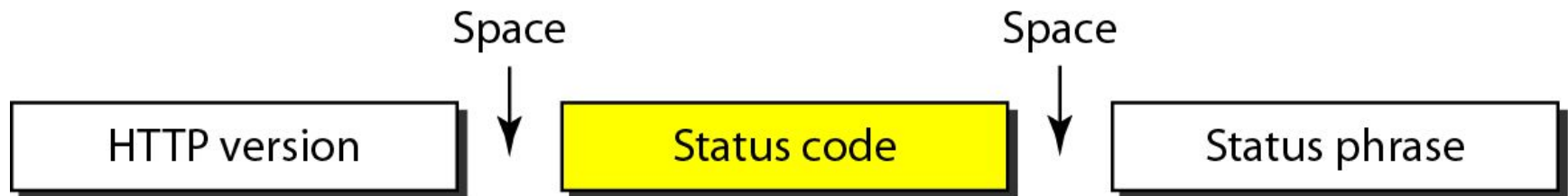
Methods

Method	Action
GET	Requests a document from the server
HEAD	Requests information about a document but not the document itself
POST	Sends some information from the client to the server
PUT	Sends a document from the server to the client
TRACE	Echoes the incoming request
CONNECT	Reserved
OPTION	Inquires about available options

Status codes

Code	Phrase	Description
Informational		
Informational		
100	Continue	The initial part of the request has been received, and the client may continue with its request.
101	Switching	The server is complying with a client request to switch protocols defined in the upgrade header.
Success		
200	OK	The request is successful.
201	Created	A new URL is created.
202	Accepted	The request is accepted, but it is not immediately acted upon.
204	No content	There is no content in the body.

UEM
UNIVERSITY OF ENGINEERING & MANAGEMENT

Contd..

Code	Phrase	Description
Redirection		
301	Moved permanently	The requested URL is no longer used by the server.
302	Moved temporarily	The requested URL has moved temporarily.
304	Not modified	The document has not been modified.
Client Error		
400	Bad request	There is a syntax error in the request.
401	Unauthorized	The request lacks proper authorization.
403	Forbidden	Service is denied.
404	Not found	The document is not found.
405	Method not allowed	The method is not supported in this URL.
406	Not acceptable	The format requested is not acceptable.
Server Error		
500	Internal server error	There is an error, such as a crash, at the server site.
501	Not implemented	The action requested cannot be performed.
503	Service unavailable	The service is temporarily unavailable, but may be requested in the future.

UEM
UNIVERSITY OF ENGINEERING & MANAGEMENT

Example

Client

Server

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

Contd..

Client

Server

Request (POST method)

POST   /cgi-bin/doc.pl HTTP/1.1
Accept: */*
Accept: image/gif
Accept: image/jpeg
Content-length: 50

(Input information)

HTTP/1.1   200 OK
Date: Mon, 07-Jan-02 13:15:14 GMT
Server: Challenger
MIME-version: 1.0
Content-length: 2000

(Body of the document)

Response

Thank You