



COMPUTER NETWORKS

Sivaraman Eswaran Ph.D.

Department of Computer Science and Engineering

COMPUTER NETWORKS

Application Layer

Sivaraman Eswaran Ph.D.

Department of Computer Science and Engineering

Unit – 2 Application Layer

2.1 Principles of Network Applications

2.2 Web, HTTP and HTTPS

2.3 The Domain Name System

2.4 P2P Applications

2.5 Socket Programming with TCP & UDP

2.6 Other Application Layer Protocols

First, a quick review...

- web page consists of *objects*, each of which can be stored on different Web servers
- object can be HTML file, JPEG image, Java applet, audio file,...
- web page consists of *base HTML-file* which includes *several referenced objects, each* addressable by a *URL*, e.g.,
- If a Web page contains HTML text and 5 JPEG images, then the Web page has 6 objects: the base HTML file plus the 5 images.

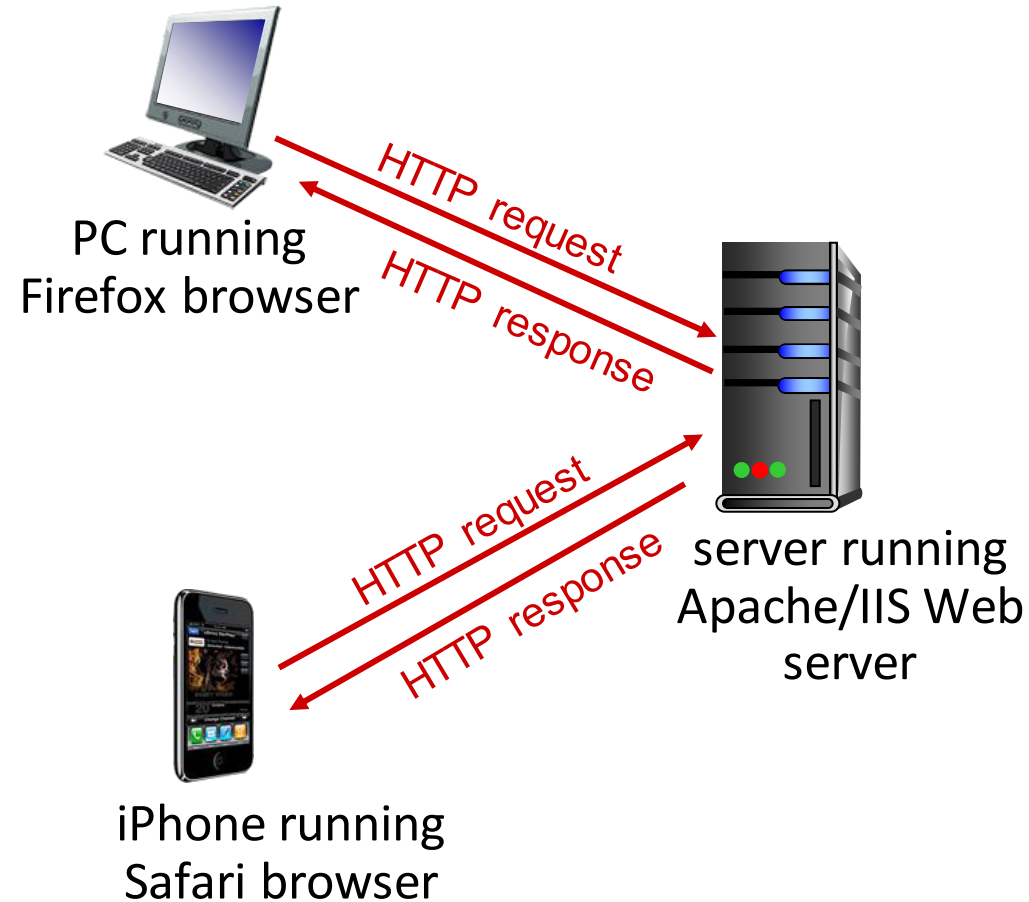
`www.someschool.edu/someDept/pic.gif`

host name

path name

HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model:
 - *client*: browser that requests, receives, (using HTTP protocol) and "displays" Web objects
 - *server*: Web server sends (using HTTP protocol) objects in response to requests



Defined in RFC 1945; RFC 2616

HTTP uses TCP:

- client initiates TCP connection (creates socket) to server, port 80
- server accepts TCP connection from client
- HTTP messages (application-layer protocol messages) exchanged between browser (HTTP client) and Web server (HTTP server)
- TCP connection closed

HTTP is “stateless”

- server maintains *no* information about past client requests

aside

protocols that maintain
“state” are complex!

- past history (state) must be maintained
- if server/client crashes, their views of “state” may be inconsistent, must be reconciled

COMPUTER NETWORKS

HTTP Connections: two types



Non-persistent HTTP

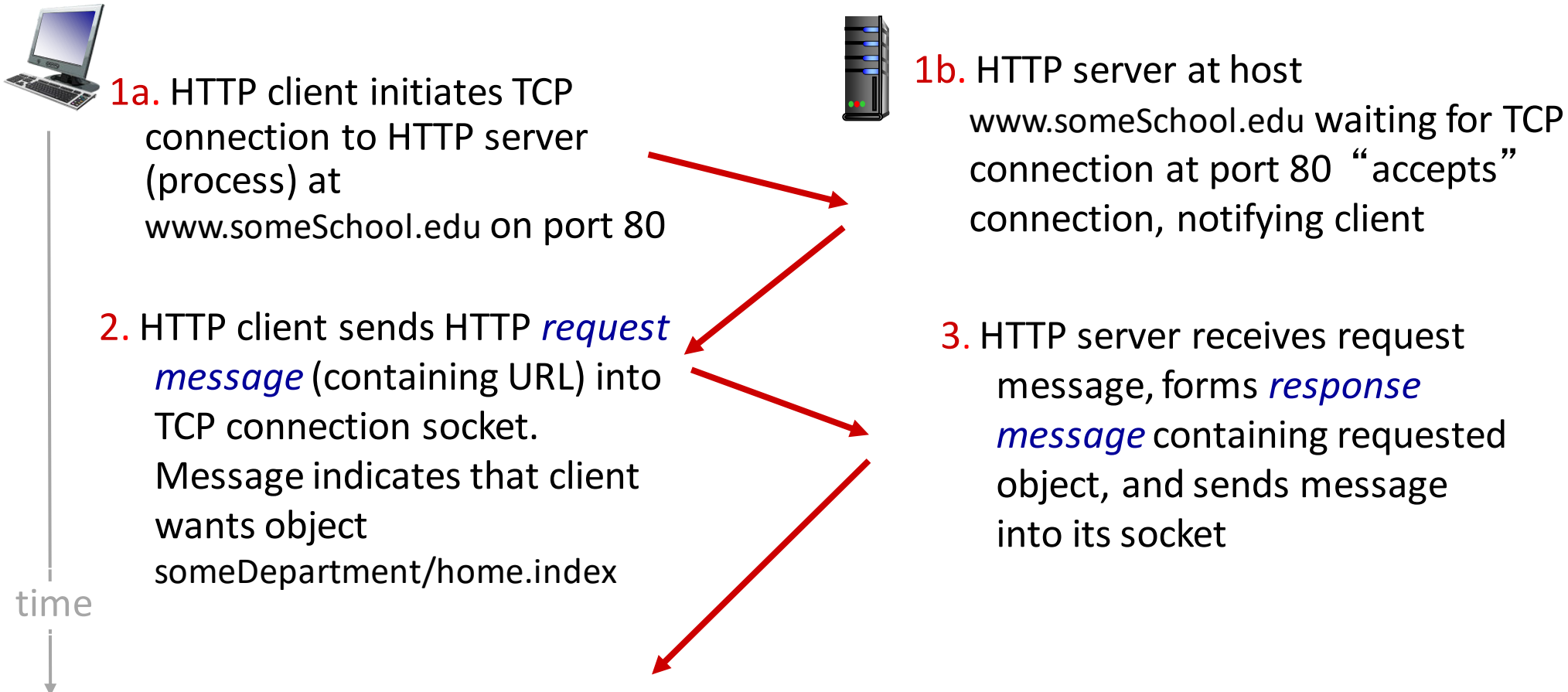
1. TCP connection opened
2. at most one object sent over TCP connection
3. TCP connection closed

downloading multiple objects
required multiple connections

Persistent HTTP

- TCP connection opened to a server
- multiple objects can be sent over *single* TCP connection between client, and that server
- TCP connection closed

User enters URL: `www.someSchool.edu/someDepartment/home.index`
(base HTML file containing text, references to 10 jpeg images)



COMPUTER NETWORKS

Non-persistent HTTP: example (more)

User enters URL: `www.someSchool.edu/someDepartment/home.index`
(containing text, references to 10 jpeg images)



4. HTTP server closes TCP connection.

5. HTTP client receives response message containing html file, displays html. Parsing html file, finds 10 referenced jpeg objects

6. Steps 1-5 repeated for each of 10 jpeg objects

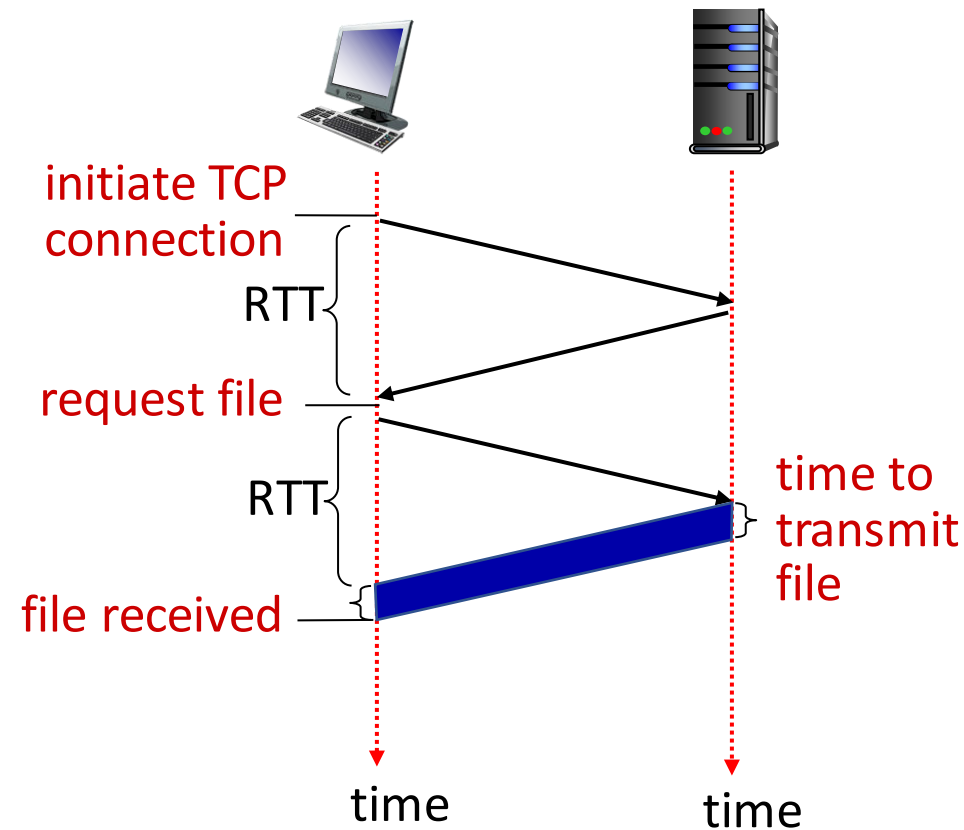
time



RTT (definition): time for a small packet to travel from client to server and back

HTTP response time (per object):

- one RTT to initiate TCP connection
- one RTT for HTTP request and first few bytes of HTTP response to return
- object/file transmission time



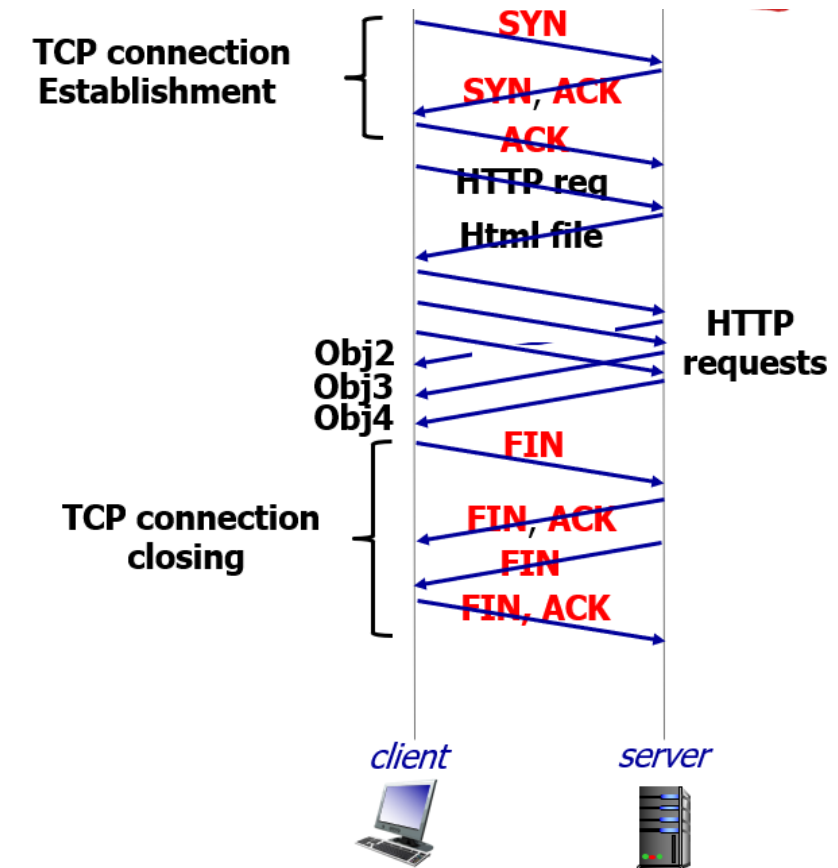
Non-persistent HTTP response time = 2RTT + file transmission time

Non-persistent HTTP issues:

- requires 2 RTTs per object
- OS overhead for *each* TCP connection (TCP buffer and variables)
- browsers often open multiple parallel TCP connections to fetch referenced objects in parallel

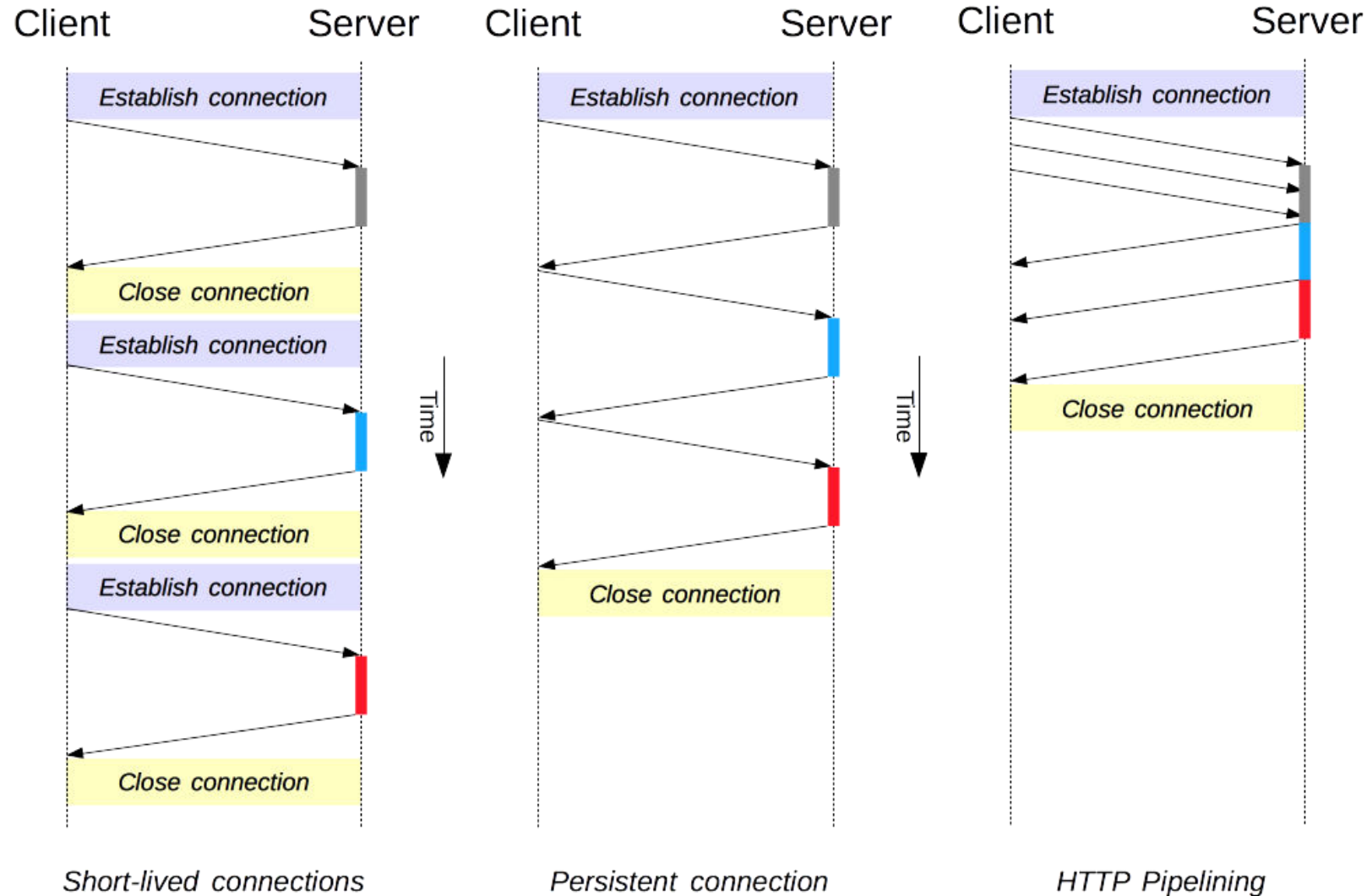
Persistent HTTP (HTTP1.1):

- server leaves connection open after sending response
- subsequent HTTP messages between same client/server sent over open connection
- client sends requests as soon as it encounters a referenced object
- as little as one RTT for all the referenced objects (cutting response time in half)



COMPUTER NETWORKS

Connection Management in HTTP/1.x





THANK YOU

Sivaraman Eswaran Ph.D.

Department of Computer Science and Engineering

sivaramane@pes.edu

+91 80 6666 3333 Extn 834