CSCI 460
Networks and Communications

Application Layer

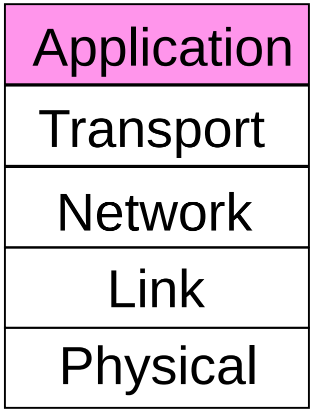
Humayun Kabir
Professor, CS, Vancouver Island University, BC, Canada

Outline

- DNS
  - Name Space
  - Resource Record
  - DNS Server
- HTTP
  - URL
  - HTML
  - HTTP Methods
  - HTTP Headers
- FTP
  - Control and Data Connections
  - Commands and Replies

The Application Layer

Uses transport services to build distributed applications



Application

Transport

Network

Link

Physical

DNS – Domain Name System

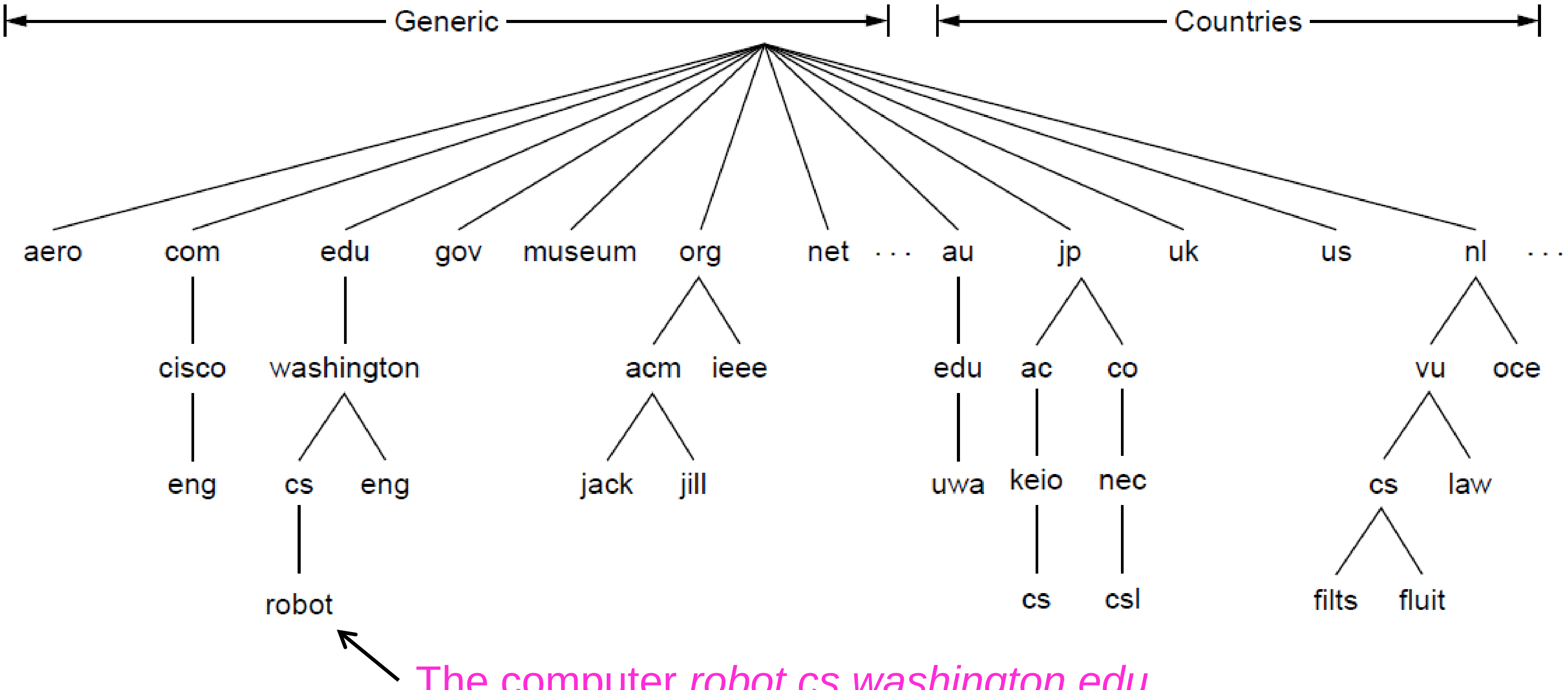
The DNS resolves high-level human readable names for computers to low-level IP addresses

- DNS name space »
- Domain Resource records »
- Name servers »

The DNS Name Space

DNS namespace is hierarchical from the root down

- Different parts delegated to different organizations



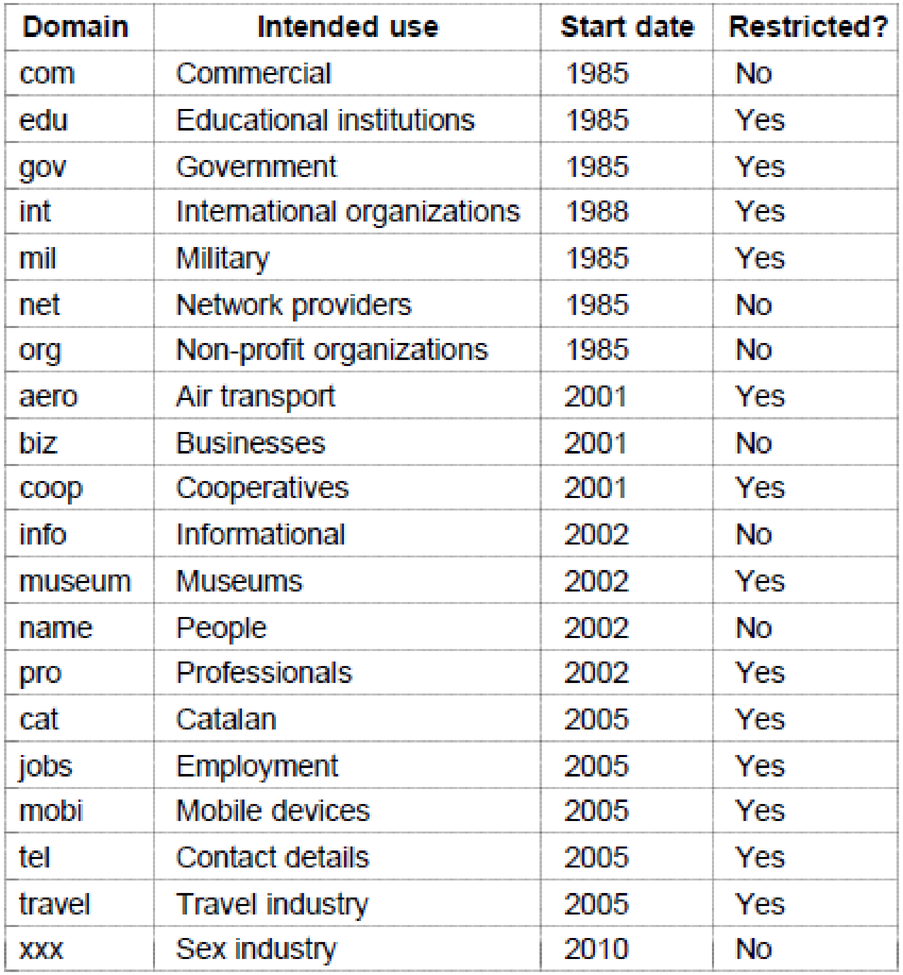
The computer *robot.cs.washington.edu*

The DNS Name Space

Generic **top-level domains** (**250**) are controlled by ICANN who appoints registrars to run them

Cybersquatting

This one was controversial



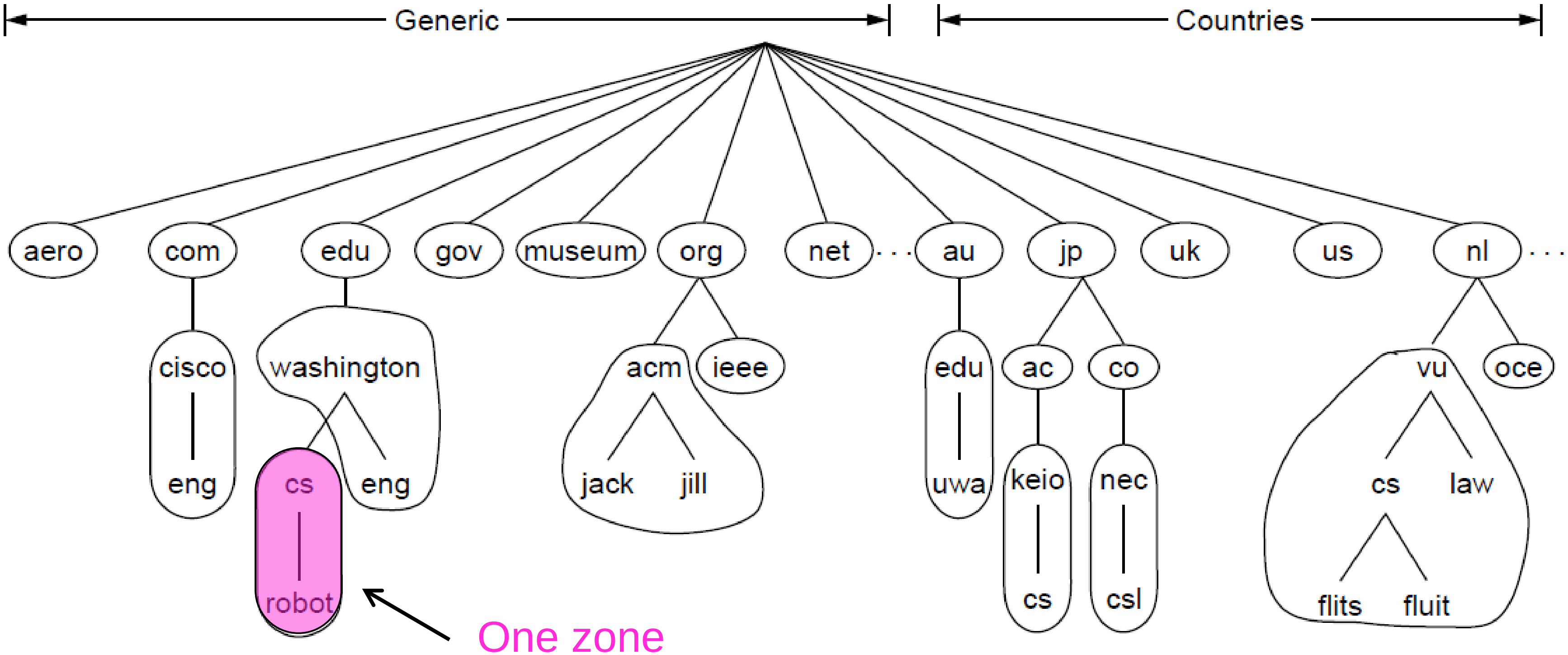
Domain	Intended use	Start date	Restricted?
com	Commercial	1985	No
edu	Educational institutions	1985	Yes
gov	Government	1985	Yes
int	International organizations	1988	Yes
mil	Military	1985	Yes
net	Network providers	1985	No
org	Non-profit organizations	1985	No
aero	Air transport	2001	Yes
biz	Businesses	2001	No
coop	Cooperatives	2001	Yes
info	Informational	2002	No
museum	Museums	2002	Yes
name	People	2002	No
pro	Professionals	2002	Yes
cat	Catalan	2005	Yes
jobs	Employment	2005	Yes
mobi	Mobile devices	2005	Yes
tel	Contact details	2005	Yes
travel	Travel industry	2005	Yes
xxx	Sex industry	2010	No

The DNS Name Space

- A **Domain Name** must be under a top level domain.
- Each domain is named by the path upward from it to the top level domain, e.g, ***eng.cisco.com***
- Each **component** in a domain name can be of **maximum 63 characters** and **case sensitive**.
- A domain name cannot have more than **255 characters in total**.

The DNS Zones

- DNS Name Space is divided into **non-overlapping zones**.



One zone

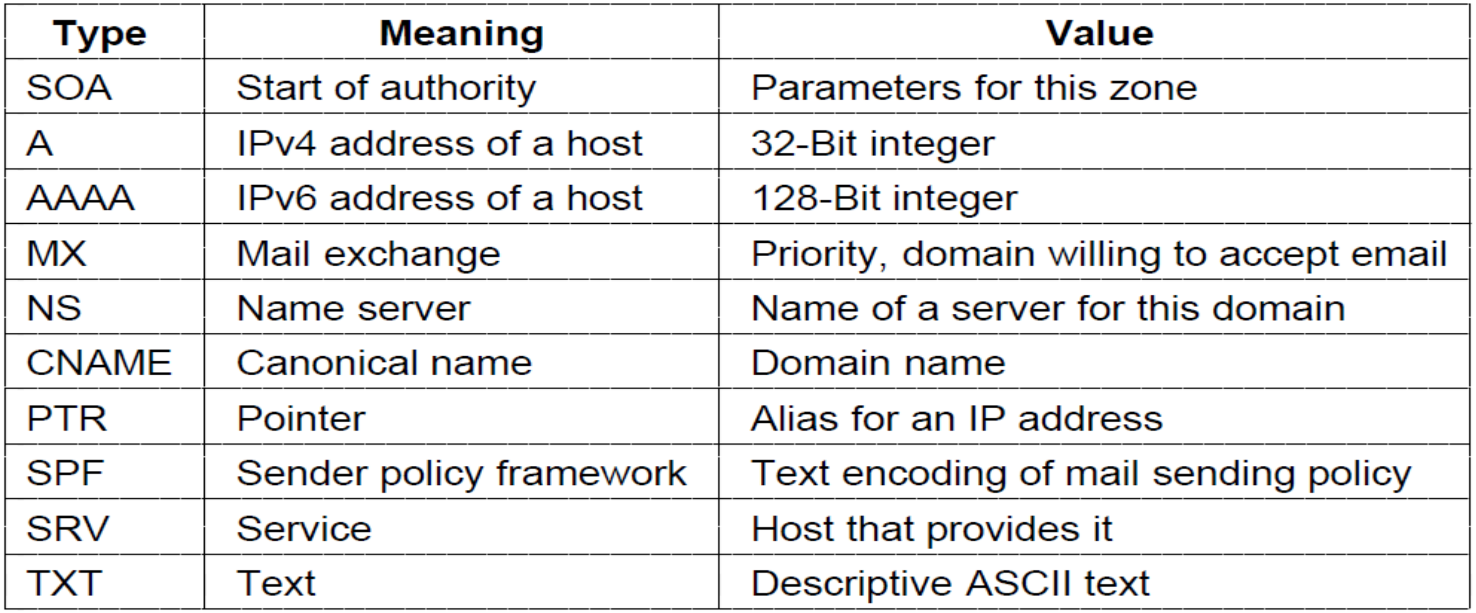
Name Servers

- Each zone is also associated with one or more **name servers**
- **Name servers** are the hosts that hold the database for the zone.
- A zone will have **one primary** name server, which gets its information from a file on its disk, and one or more **secondary** name servers, which get their information from the primary name server.
- Zone information in the name servers are kept as a **zone file** that holds many **resource records** about the zone.

Domain Resource Records

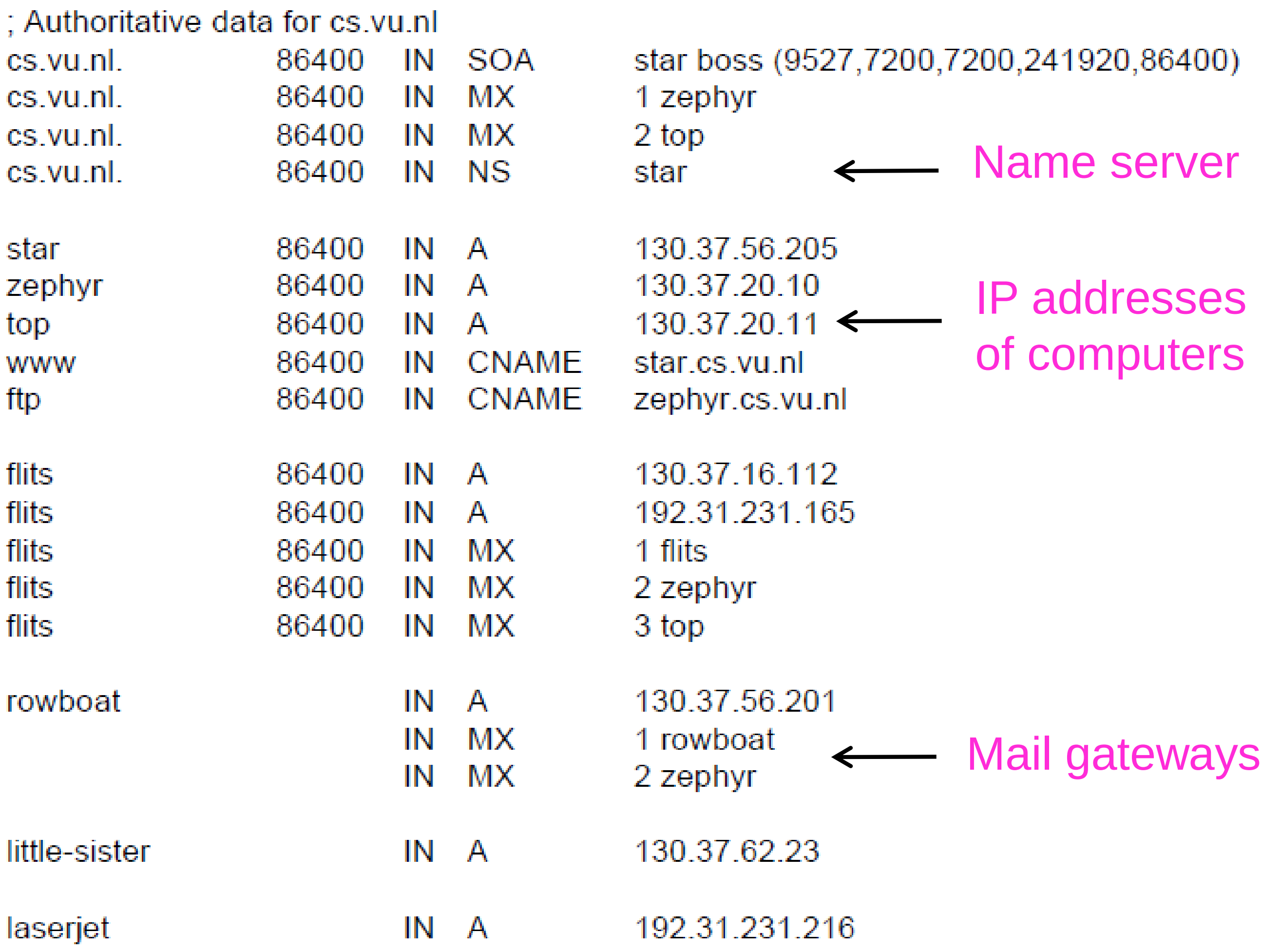
Domain Name	Time to live	Class	Type	Value
cs.vu.nl	86400	IN	NS	star
star	86400	IN	A	130.37.56.203
www	86400	IN	CNAME	star.cs.vu.nl

The key resource records are IP addresses (A or AAAA) and name servers (NS)



Type	Meaning	Value
SOA	Start of authority	Parameters for this zone
A	IPv4 address of a host	32-Bit integer
AAAA	IPv6 address of a host	128-Bit integer
MX	Mail exchange	Priority, domain willing to accept email
NS	Name server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
SPF	Sender policy framework	Text encoding of mail sending policy
SRV	Service	Host that provides it
TXT	Text	Descriptive ASCII text

Domain Resource Records



```
; Authoritative data for cs.vu.nl
cs.vu.nl.        86400   IN   SOA      star boss (9527,7200,7200,241920,86400)
cs.vu.nl.        86400   IN   MX       1 zephyr
cs.vu.nl.        86400   IN   MX       2 top
cs.vu.nl.        86400   IN   NS       star

star             86400   IN   A        130.37.56.205
zephyr           86400   IN   A        130.37.20.10
top              86400   IN   A        130.37.20.11
www              86400   IN   CNAME    star.cs.vu.nl
ftp              86400   IN   CNAME    zephyr.cs.vu.nl

flits            86400   IN   A        130.37.16.112
flits            86400   IN   A        192.31.231.165
flits            86400   IN   MX       1 flits
flits            86400   IN   MX       2 zephyr
flits            86400   IN   MX       3 top

rowboat                  IN   A        130.37.56.201
                         IN   MX       1 rowboat
                         IN   MX       2 zephyr

little-sister            IN   A        130.37.62.23

laserjet                 IN   A        192.31.231.216
```

Name server

IP addresses of computers

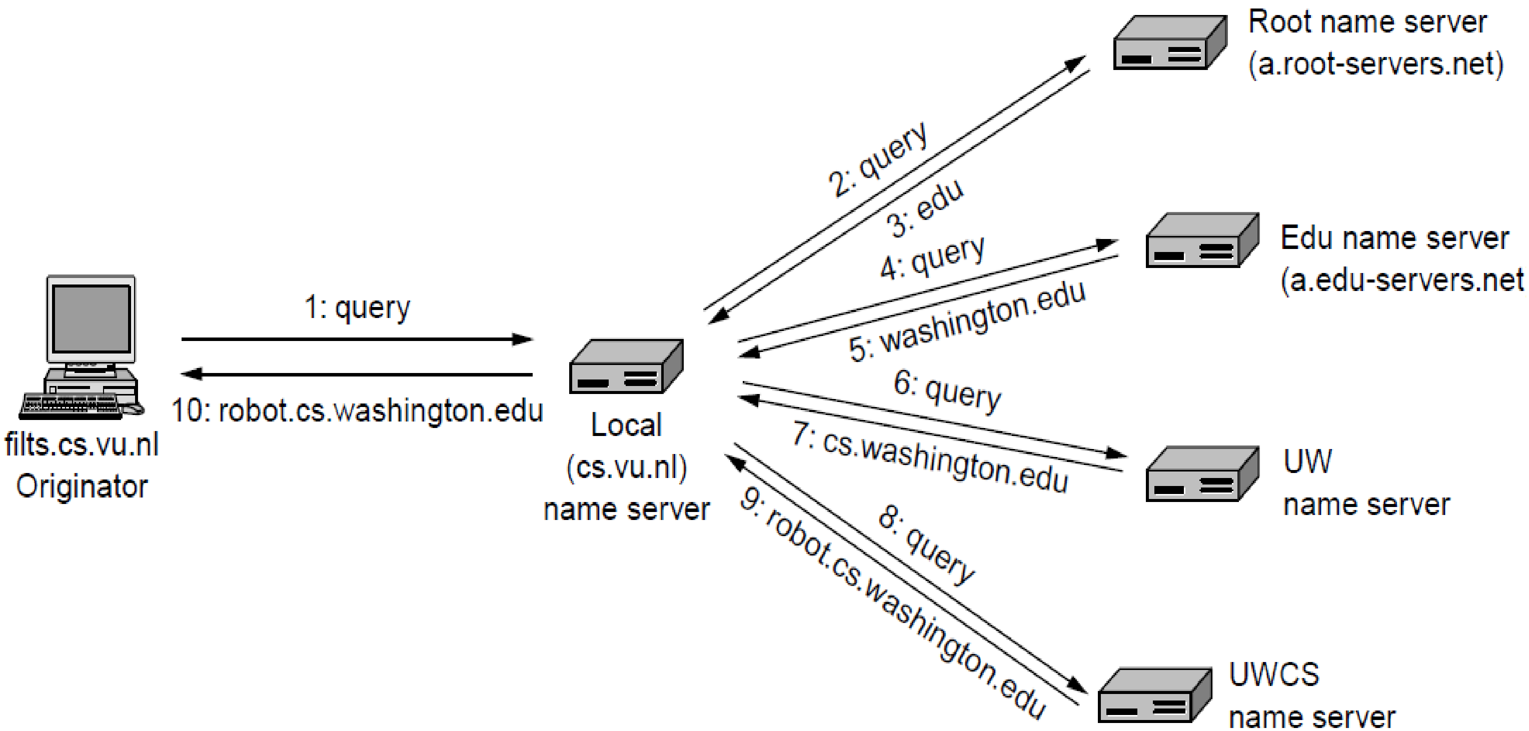
Mail gateways

A portion of imaginary DNS database for cs.vu.nl.

Name Resolution

Finding the IP address for a given hostname is called name resolution and is done with the **DNS protocol**.

- Runs on UDP port 53, retransmits lost messages
- Caches name server answers for better performance



Name Resolution

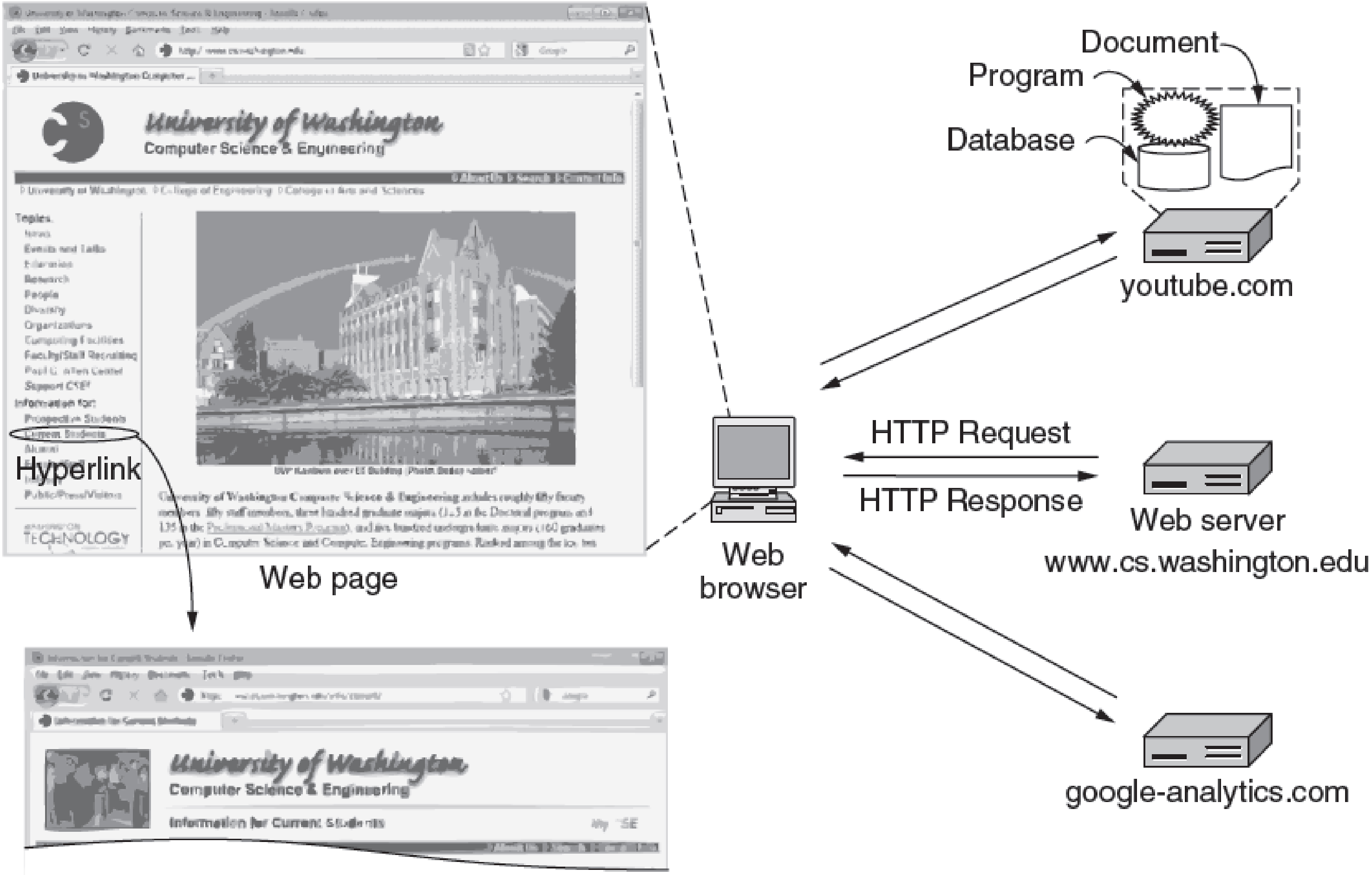
- Computer requests a local name server to resolve IP address against a domain name.
- Local name server attempts to resolve from the cache.
- If cache attempt fails, local name server queries (**recursive query**) the root name server.
- Root returns the name server address of the top level domain.
- Local name server queries (**iterative query**) the top level domain name server.
- Top level domain name server returns the name server address of the next level name server.
- **Iterative queries** at local name server continues until a remote name server responds with an authoratitive answer.

World Wide Web (HTTP)

- HTTP (**HyperText Transfer Protocol**) is a request-response protocol that runs on top of **TCP**.
- HTTP Client (**Web Browser**) sends the requests to HTTP Server (**Web Server**).
- HTTP Server responds to HTTP Client's requests.
- Client fetches web pages from server.
- Server usually runs on port 80.
- Headers are given in readable ASCII.
- Content is described with MIME types.
- Protocol has support for pipelining requests.
- Protocol has support for caching.

WWW Architectural Overview

HTTP transfers pages from servers to browsers

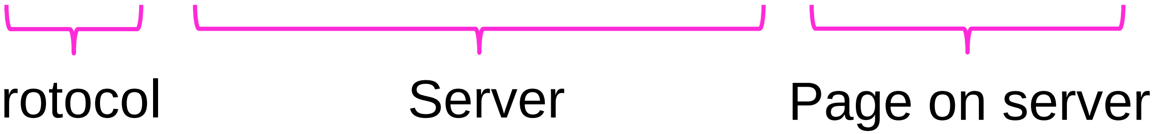


CN5E by Tanenbaum & Wetherall, © Pearson Education-Prentice Hall and D. Wetherall, 2011

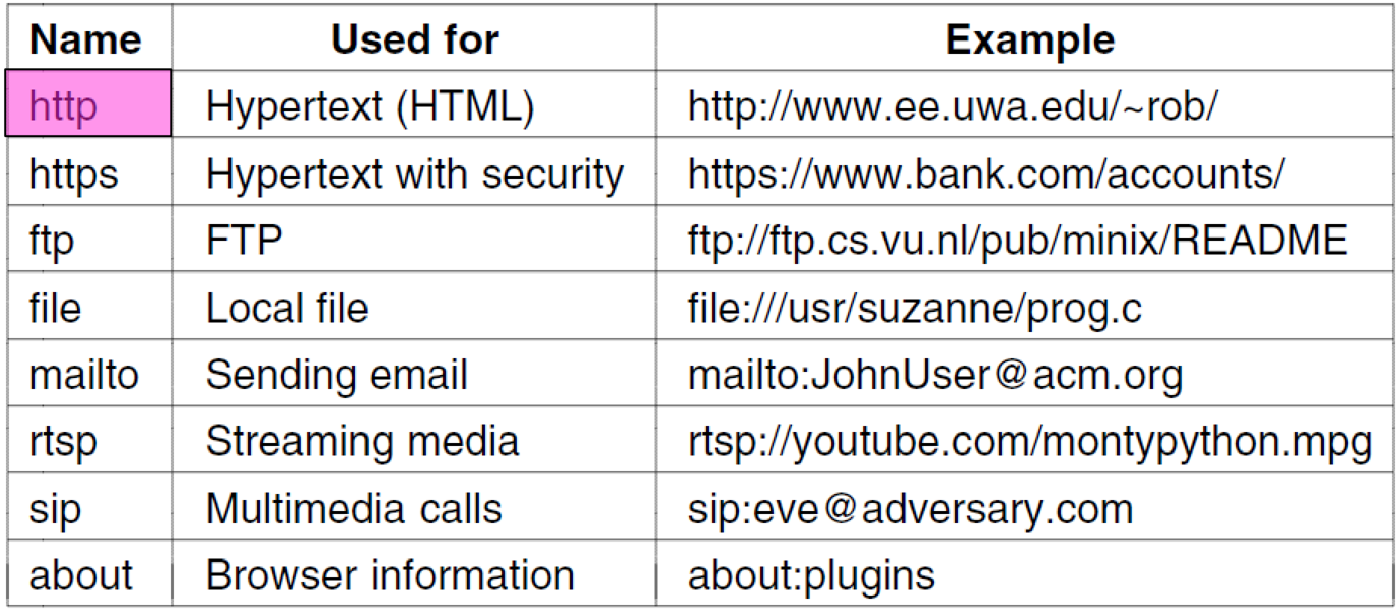
WWW Architectural Overview

Pages are named with URLs (Uniform Resource Locators)

- Example: http://www.phdcomics.com/comics.php



Our focus →



Name	Used for	Example
http	Hypertext (HTML)	http://www.ee.uwa.edu/~rob/
https	Hypertext with security	https://www.bank.com/accounts/
ftp	FTP	ftp://ftp.cs.vu.nl/pub/minix/README
file	Local file	file:///usr/suzanne/prog.c
mailto	Sending email	mailto:JohnUser@acm.org
rtsp	Streaming media	rtsp://youtube.com/montypython.mpg
sip	Multimedia calls	sip:eve@adversary.com
about	Browser information	about:plugins

Common URL protocols

WWW Architectural Overview

Steps a client (browser) takes to follow a hyperlink:

- Determine the protocol (HTTP or HTTPS)
- Ask DNS for the IP address of the server
- Make a TCP connection to the server
- Send request for the page; server sends it back
- Fetch other URLs as needed to display the page
- Close idle TCP connections

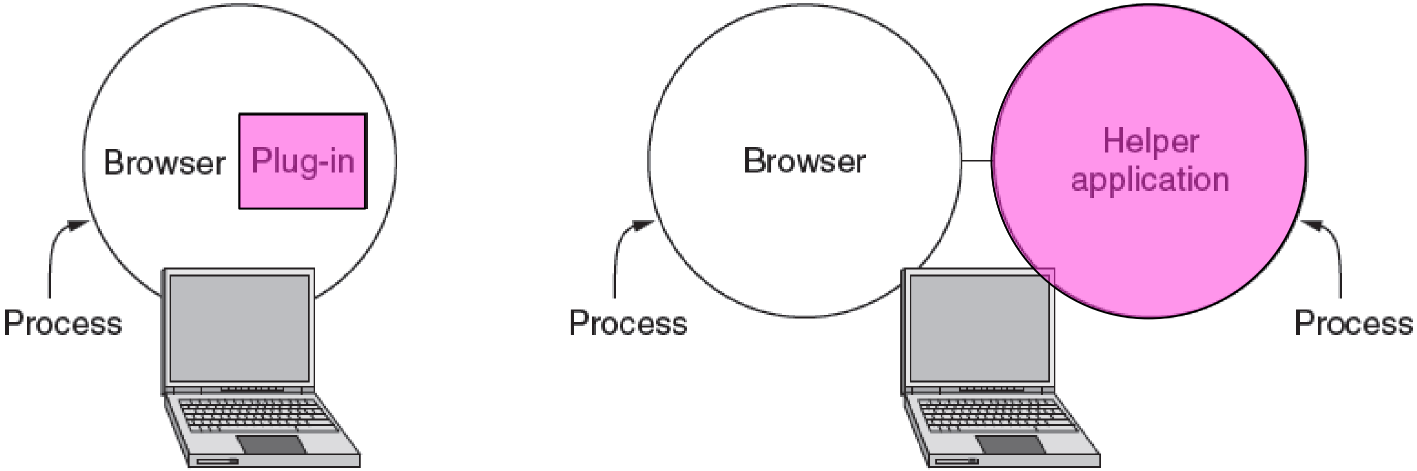
Steps a server takes to serve pages:

- Accept a TCP connection from client
- Get page request and map it to a resource (e.g., file name)
- Get the resource (e.g., file from disk)
- Send contents of the resource to the client.
- Release idle TCP connections

WWW Architectural Overview

Content type is identified by MIME types

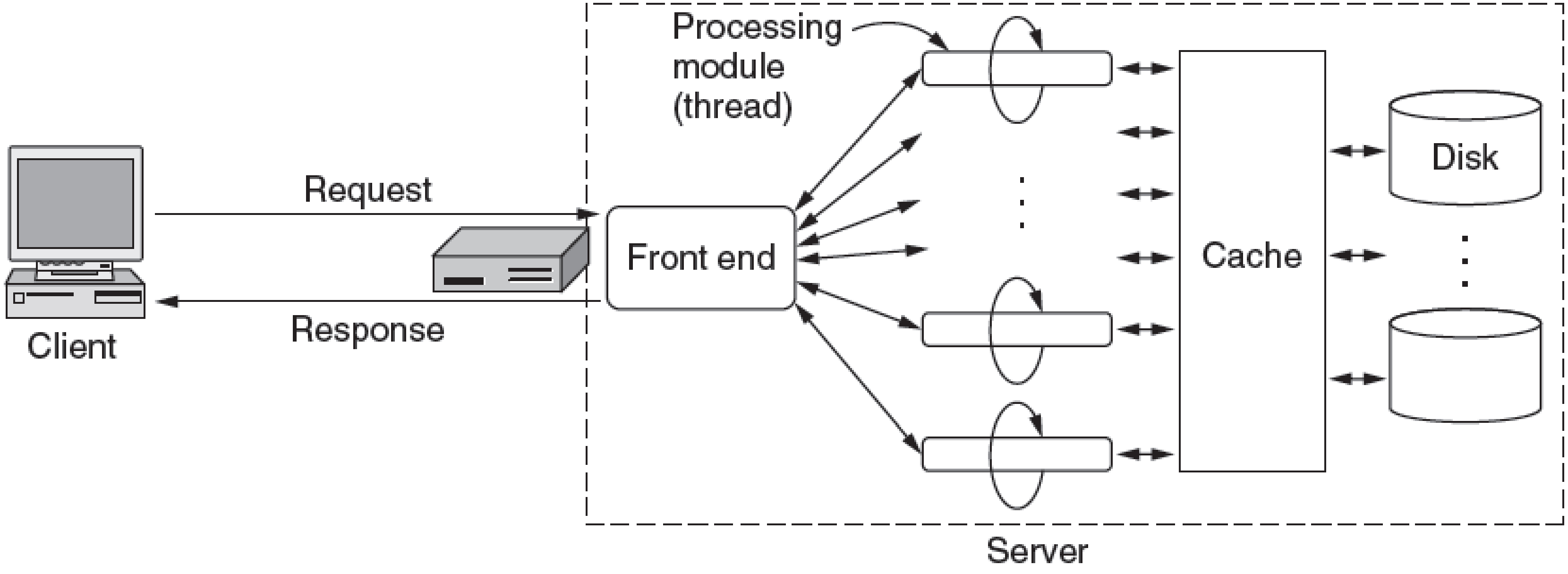
- Browser takes the appropriate action to display
- Plug-ins / helper apps extend browser for new types



WWW Architectural Overview

To scale performance, Web servers can use:

- Caching, multiple threads, and a front end



WWW Architectural Overview

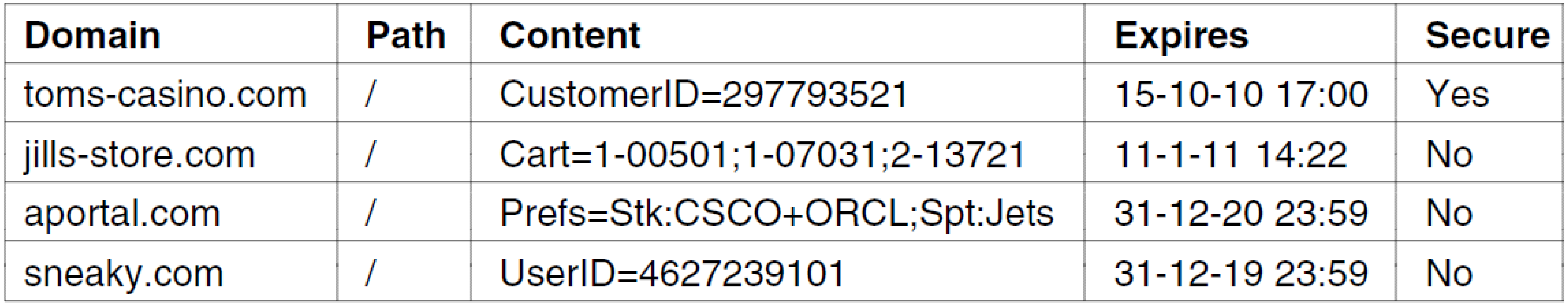
Server steps, revisited:

- Resolve name of Web page requested
- Perform access control on the Web page
- Check the cache
- Fetch requested page from disk or run program, if necassury
- Determine the rest of the response
- Return the response to the client
- Make an entry in the server log

WWW Stateful Client-Server Interactions

Cookies support stateful client/server interactions

- Server sends cookies (state) in **Set-Cookie header** with page response
- Client stores cookies in its cache across page fetches
- Client sends cookies in **Cookie header** back to the server with successive requests

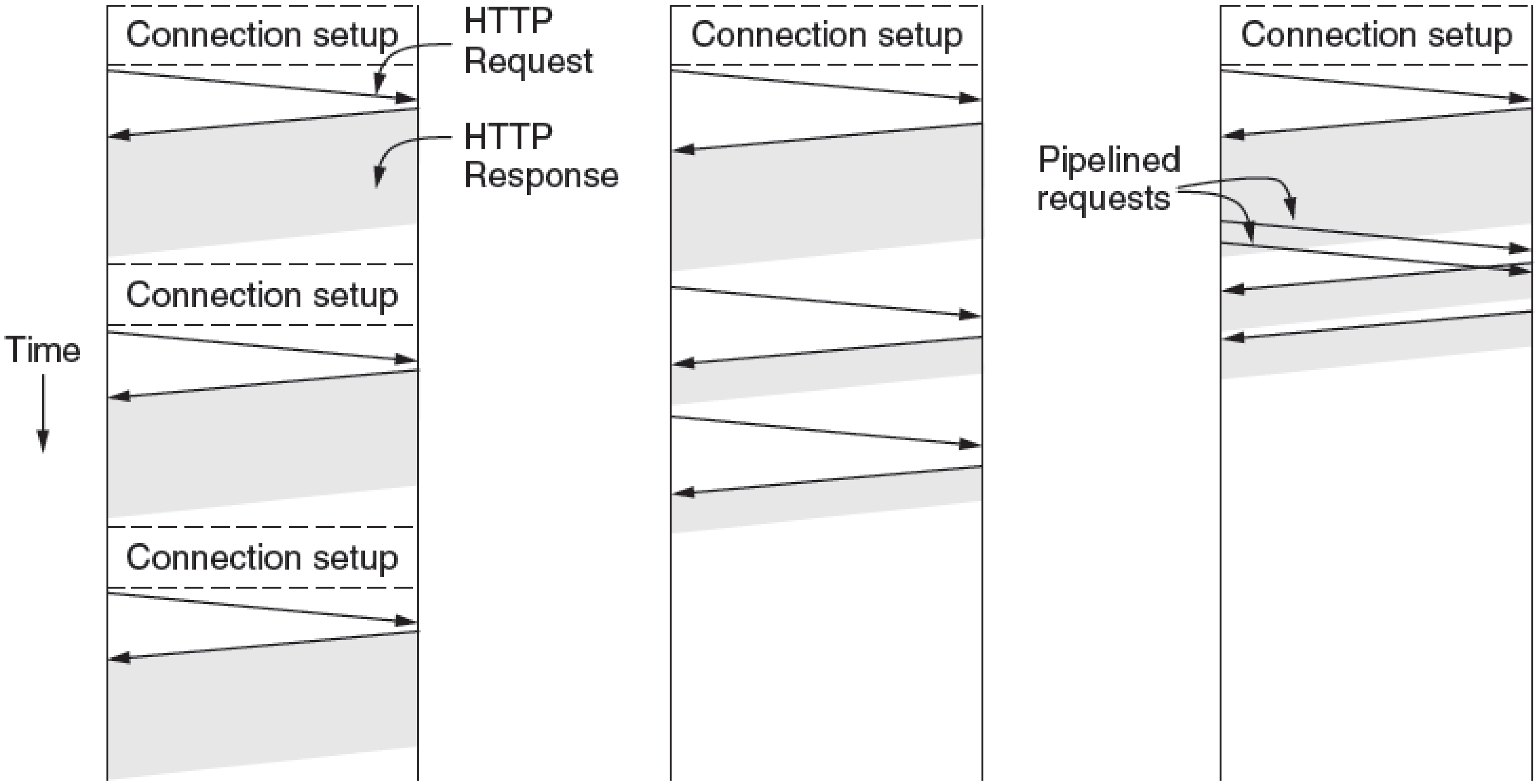


Domain	Path	Content	Expires	Secure
toms-casino.com	/	CustomerID=297793521	15-10-10 17:00	Yes
jills-store.com	/	Cart=1-00501;1-07031;2-13721	11-1-11 14:22	No
aportal.com	/	Prefs=Stk:CSCO+ORCL;Spt:Jets	31-12-20 23:59	No
sneaky.com	/	UserID=4627239101	31-12-19 23:59	No

Examples of cookies

HTTP on TCP

HTTP uses persistent connections to improve performance



One connection for each request

Sequential requests on one connection

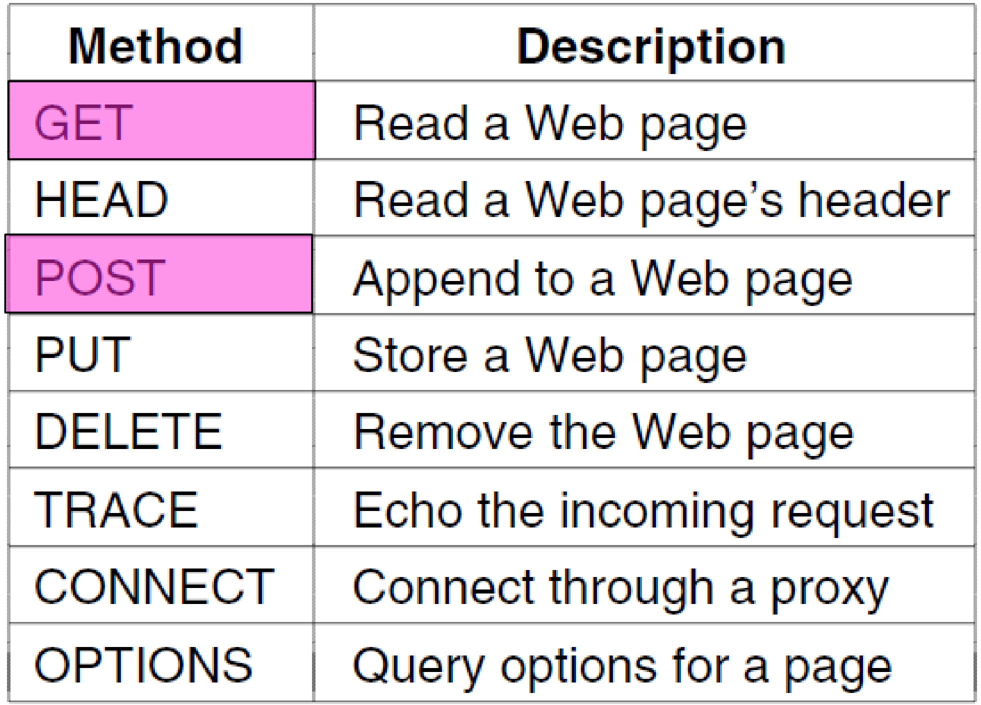
Pipelined requests on one connection

HTTP Request Methods

HTTP has several request methods.

Fetch a page → GET

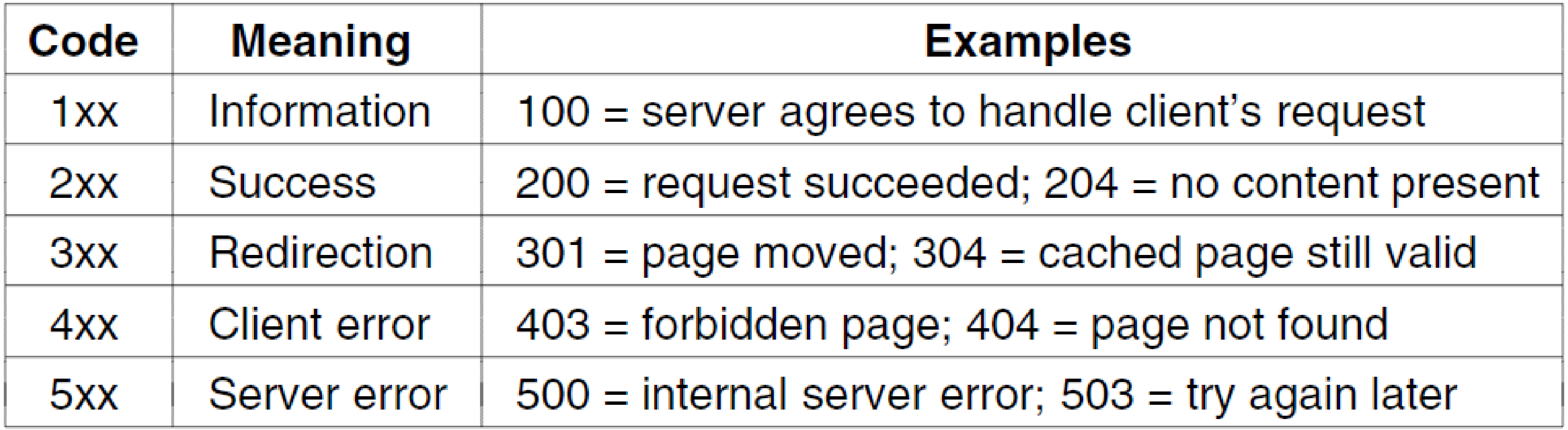
Used to send input data to a server program → POST



Method	Description
GET	Read a Web page
HEAD	Read a Web page’s header
POST	Append to a Web page
PUT	Store a Web page
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Connect through a proxy
OPTIONS	Query options for a page

HTTP Response Codes

Response codes tell the client how the request fared:



Code	Meaning	Examples
1xx	Information	100 = server agrees to handle client's request
2xx	Success	200 = request succeeded; 204 = no content present
3xx	Redirection	301 = page moved; 304 = cached page still valid
4xx	Client error	403 = forbidden page; 404 = page not found
5xx	Server error	500 = internal server error; 503 = try again later

HTTP Headers

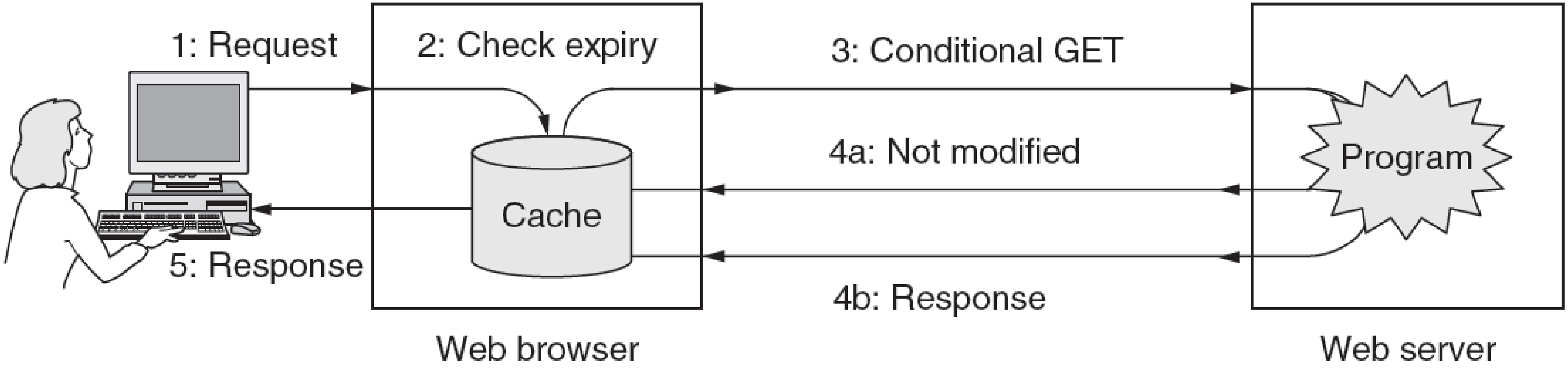
Many headers carry key information:

Function	**Example Headers**
Browser capabilities (client → server)	User-Agent, Accept, Accept-Charset, Accept-Encoding, Accept-Language
Caching related (mixed directions)	If-Modified-Since, If-None-Match, Date, Last-Modified, Expires, Cache-Control, ETag
Browser context (client → server)	Cookie, Referrer, Authorization, Host
Content delivery (server → client)	Content-Encoding, Content-Length, Content-Type, Content-Language, Content-Range, Set-Cookie

Browser Caching

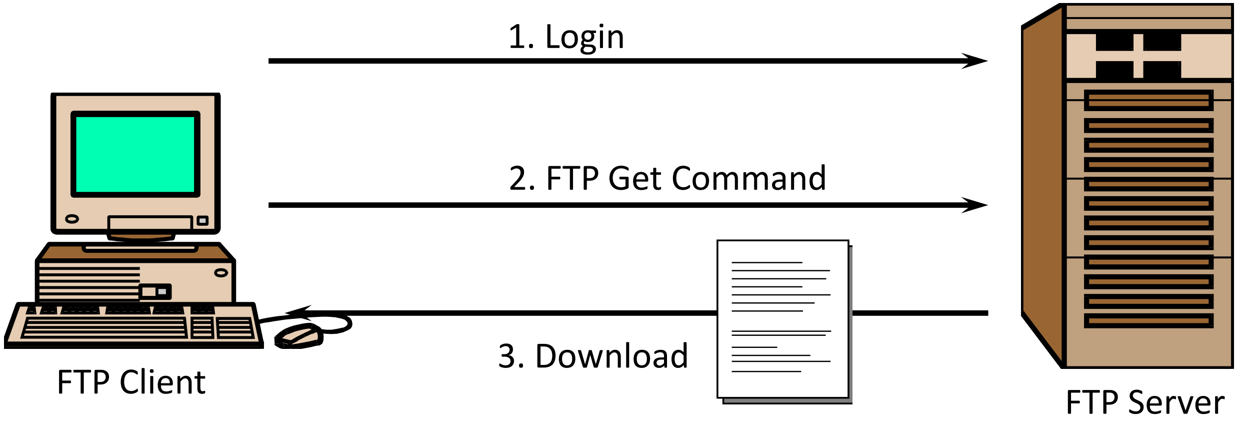
HTTP caching checks to see if the browser has a known fresh copy, and if not if the server has updated the page

- Uses a collection of headers for the checks
- Can include further levels of caching (e.g., proxy)



File Transfer Protocol (FTP)

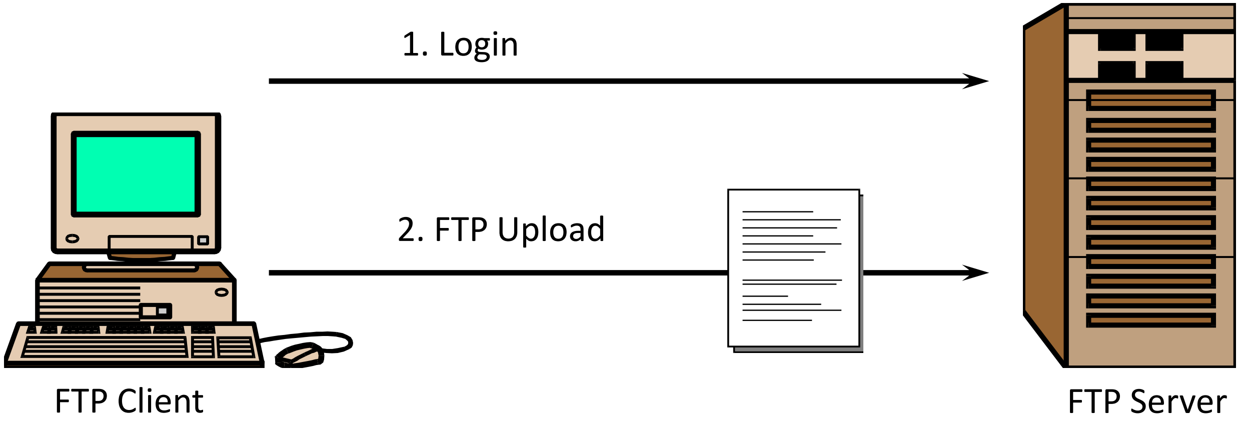
- FTP server stores files
- Client logs into host
- Client program sends command to get a file
- FTP client downloads the file with error correction



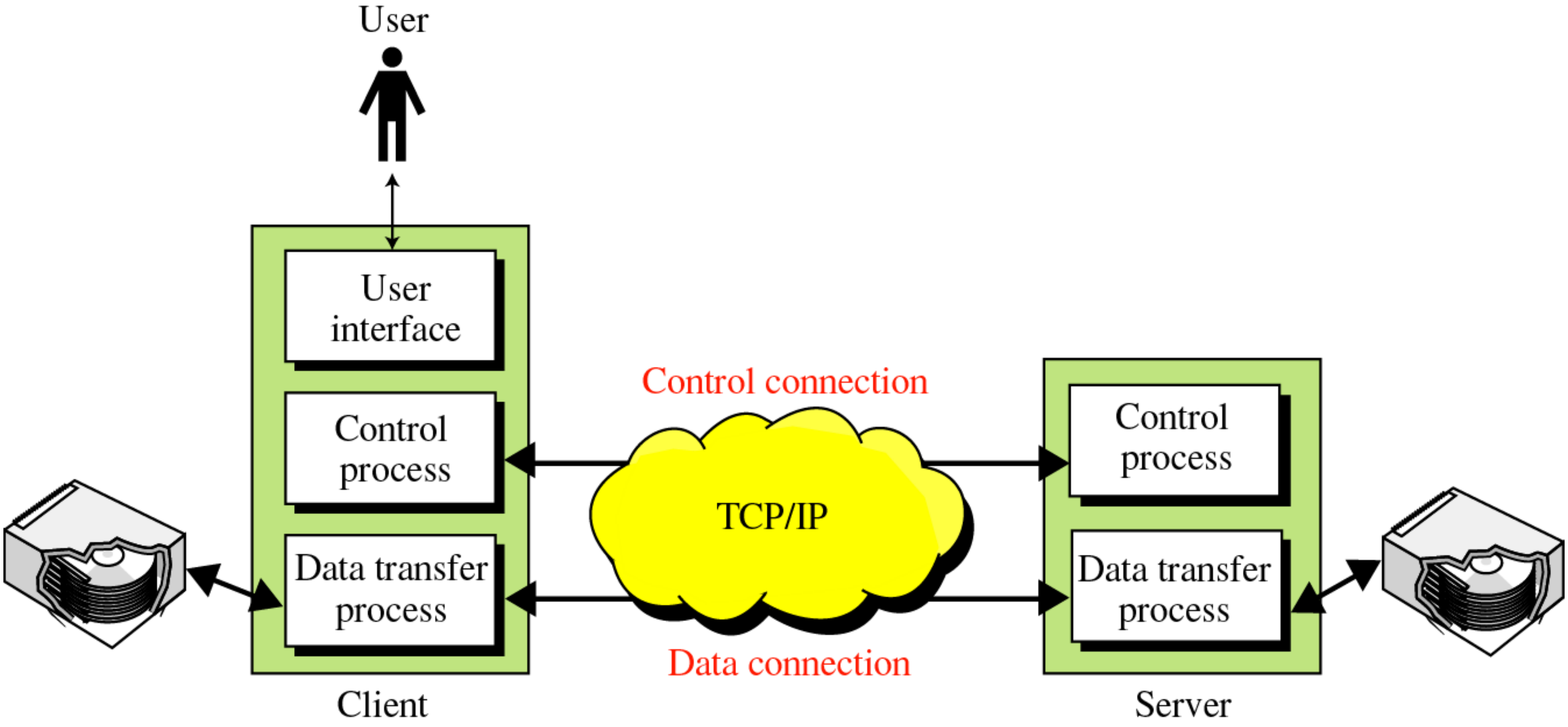
File Transfer Protocol (FTP)

User can also *upload* a file to the FTP Server

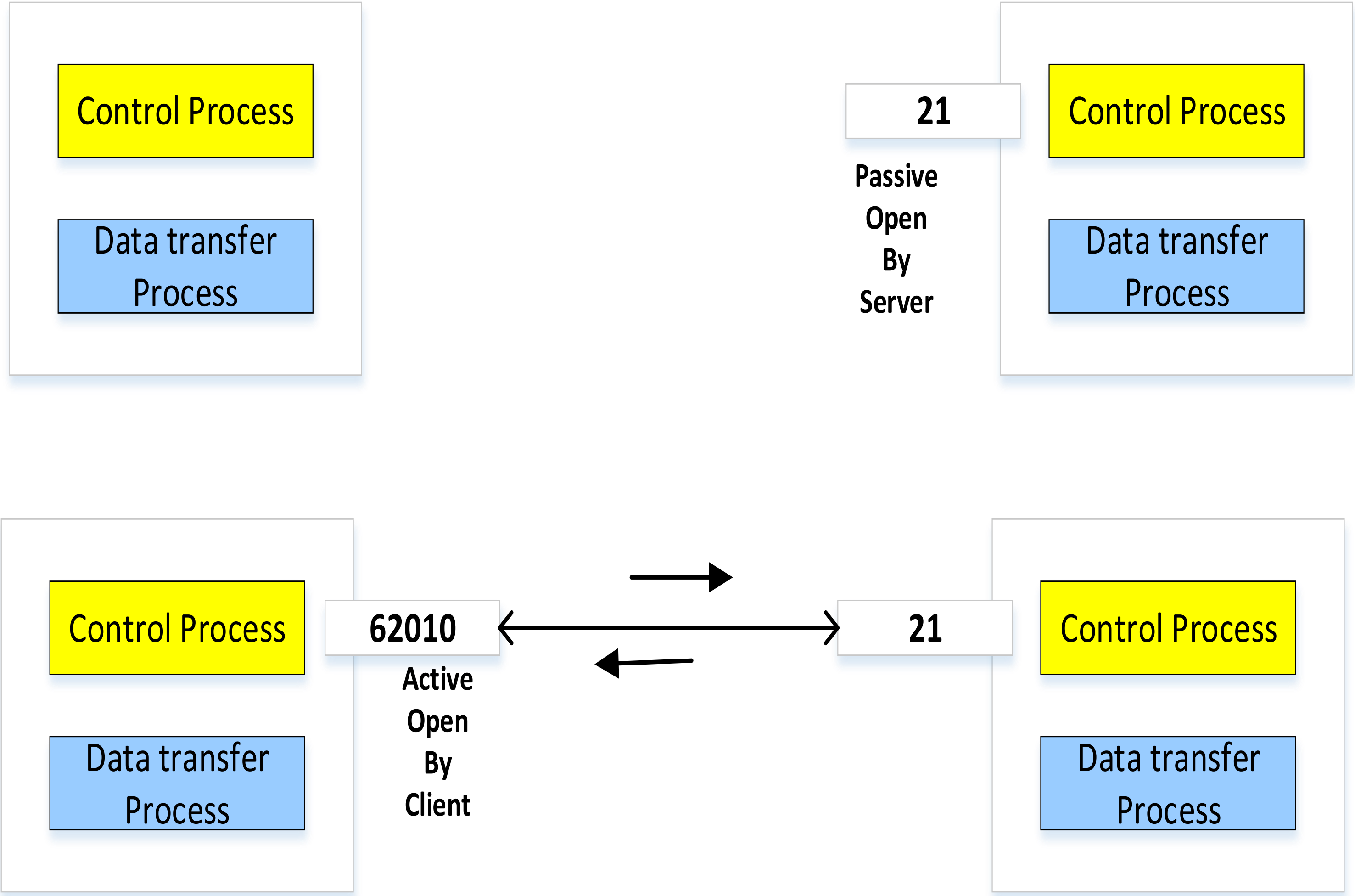
- WWW cannot do this



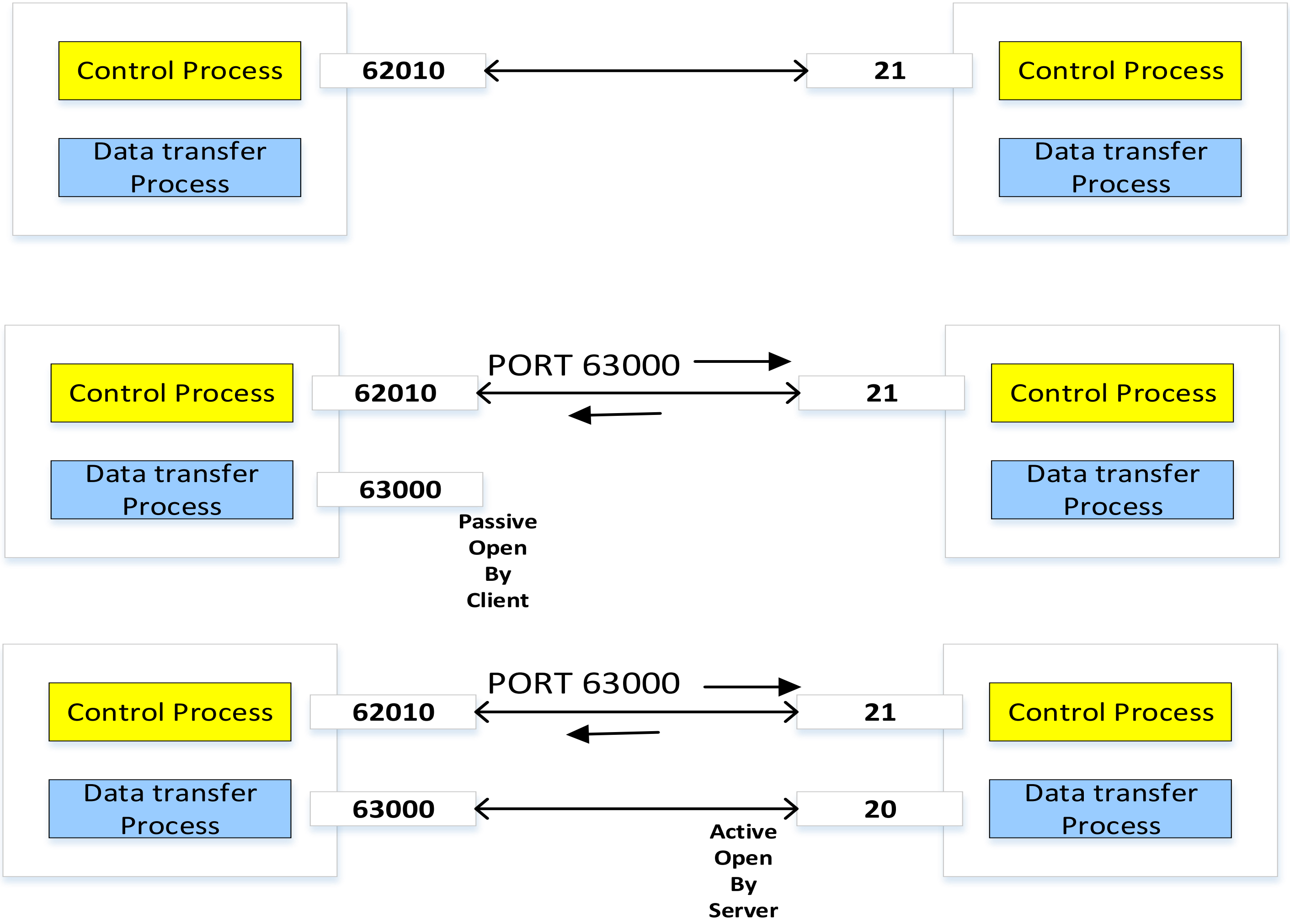
File Transfer Protocol (FTP)



FTP Control Connection



FTP Data Connection: Active Mode



Control Process

62010

21

Control Process

Data transfer Process

Data transfer Process

Control Process

62010

PORT 63000

21

Control Process

Data transfer Process

63000

Passive Open By Client

Data transfer Process

Control Process

62010

PORT 63000

21

Control Process

Data transfer Process

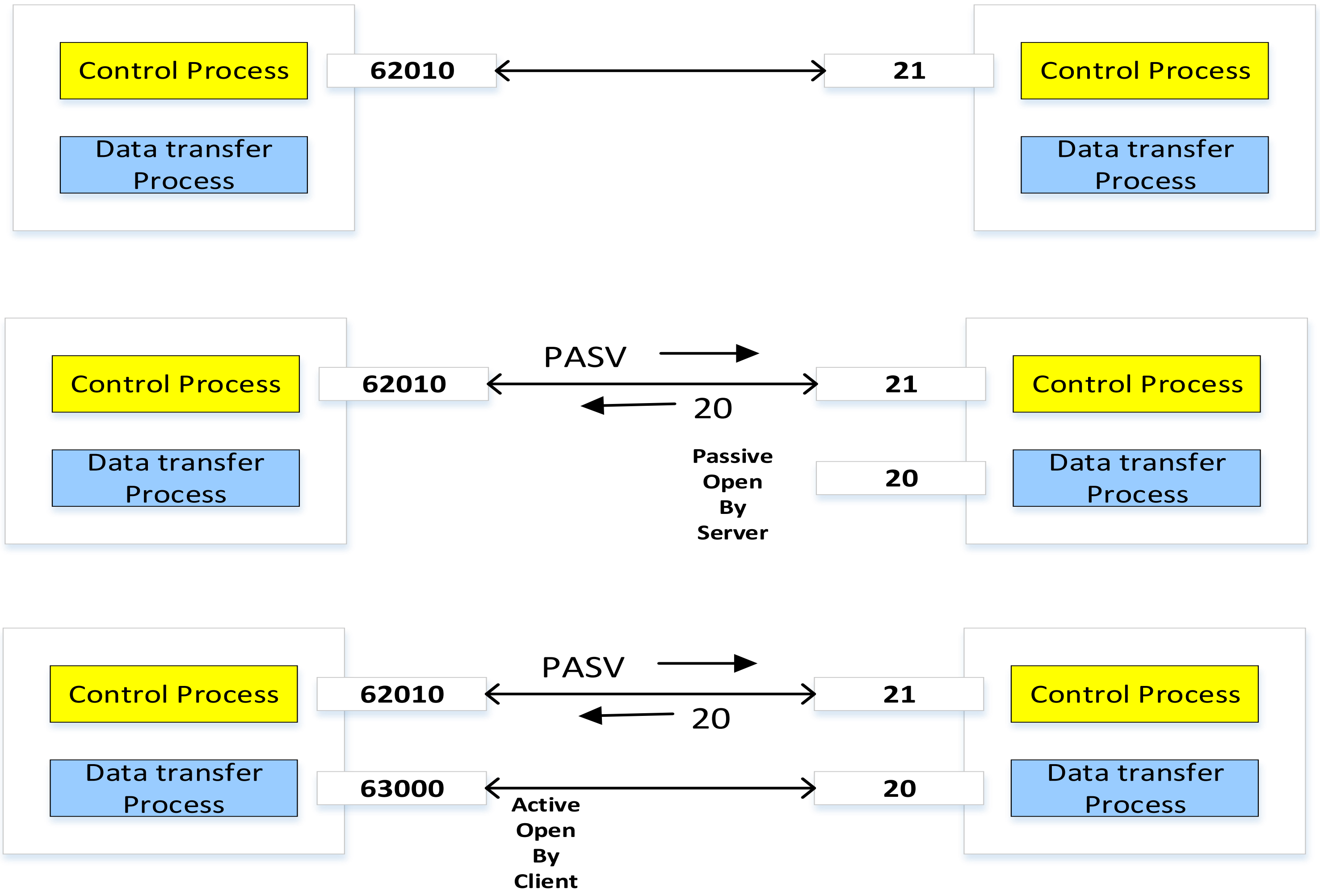
63000

20

Data transfer Process

Active Open By Server

FTP Data Connection: Passive Mode

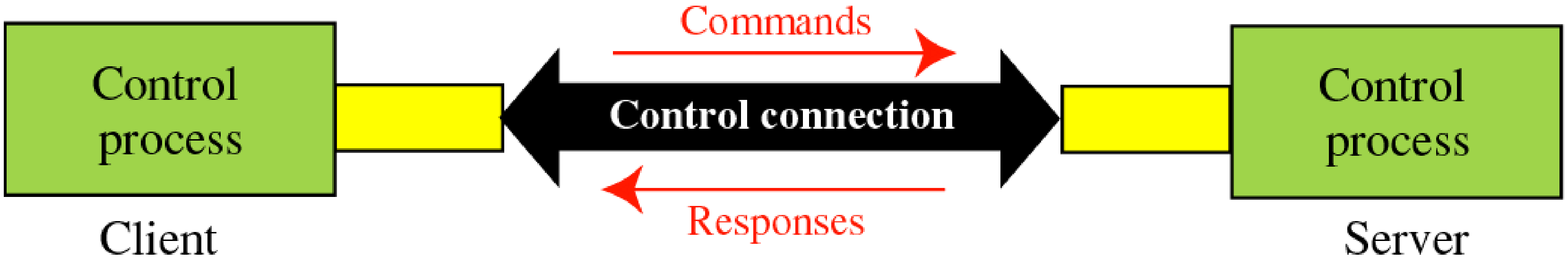


Control Process
62010
21
Control Process
Data transfer Process
Data transfer Process

Control Process
62010
PASV
21
Control Process
20
Passive Open By Server
Data transfer Process
20
Data transfer Process

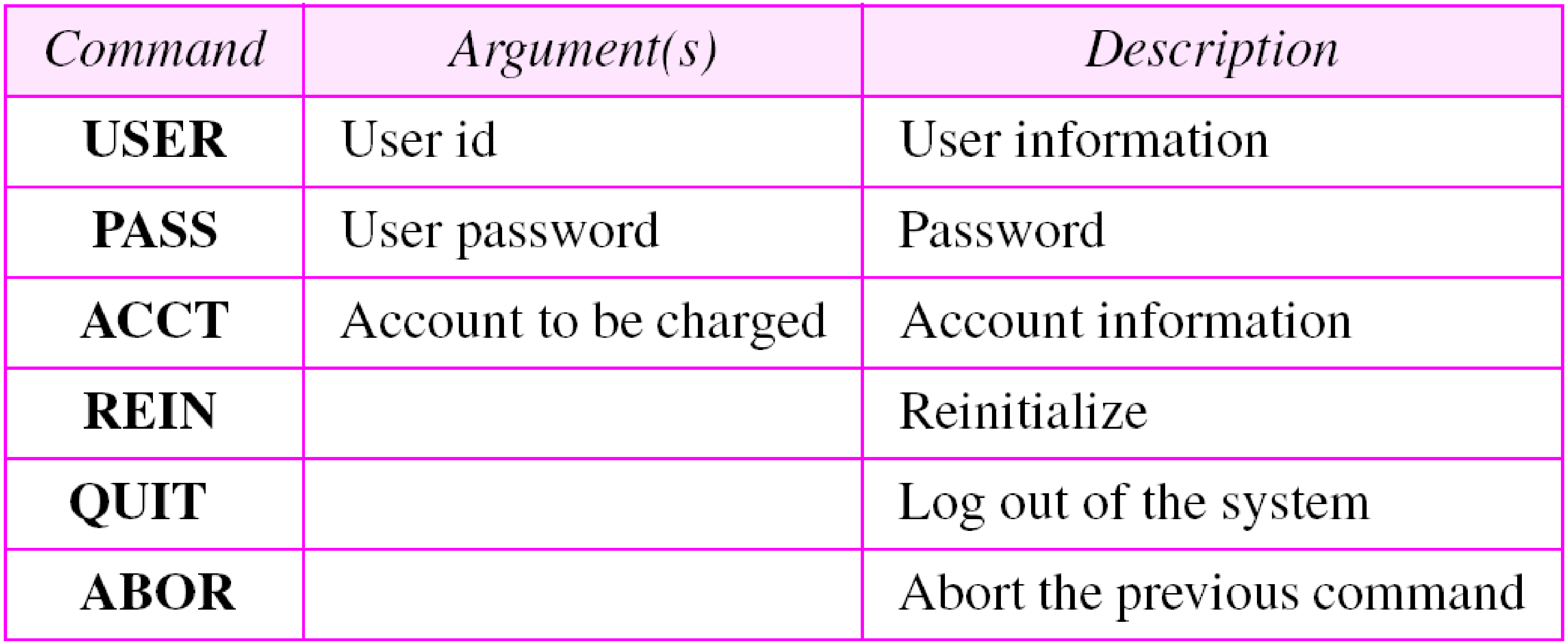
Control Process
62010
PASV
21
Control Process
20
Data transfer Process
63000
20
Data transfer Process
Active Open By Client

FTP Commands



- Access Commands
- File Management
- Data Formatting
- Port defining
- File transfer
- Miscellaneous

FTP Access Commands

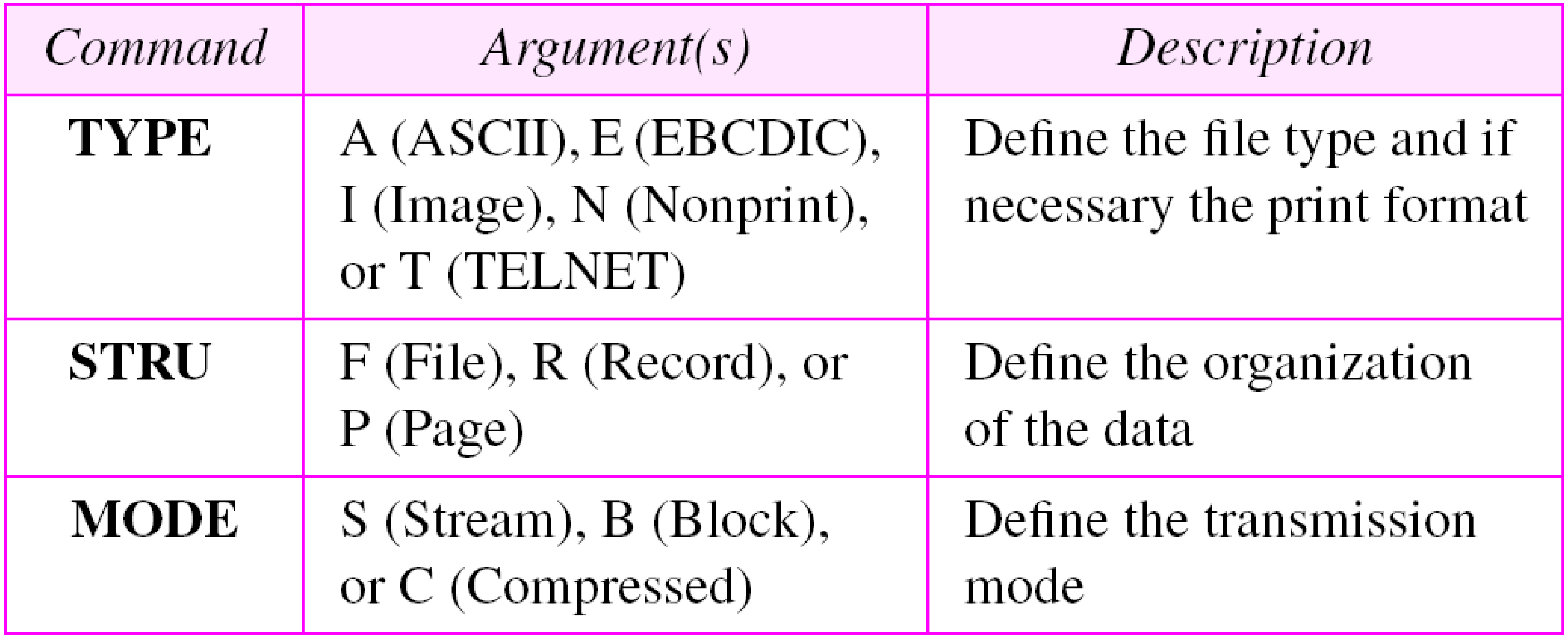


Command	*Argument(s)*	*Description*
USER	User id	User information
PASS	User password	Password
ACCT	Account to be charged	Account information
REIN		Reinitialize
QUIT		Log out of the system
ABOR		Abort the previous command

FTP File Management Commands

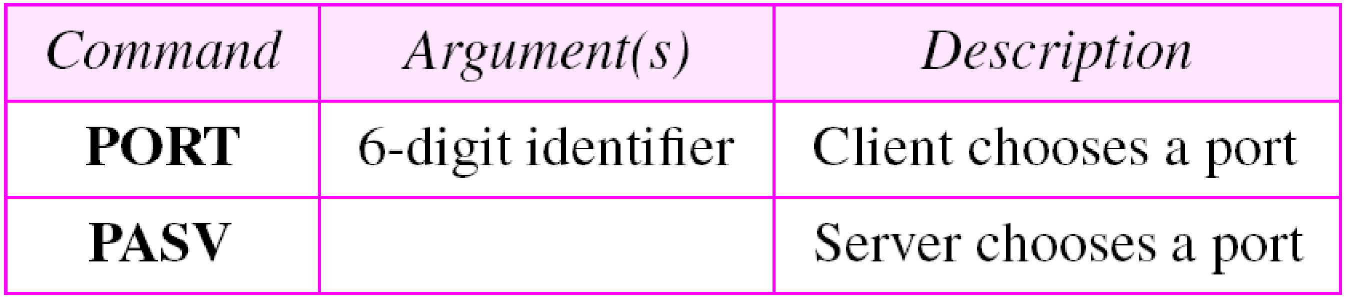
Command	*Argument(s)*	*Description*
CWD	Directory name	Change to another directory
CDUP		Change to the parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
NLIST	Directory name	List the names of subdirectories or files without other attributes
MKD	Directory name	Create a new directory
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old file name)	Identify a file to be renamed
RNTO	File name (new file name)	Rename the file
SMNT	File system name	Mount a file system

FTP Data Formatting Commands



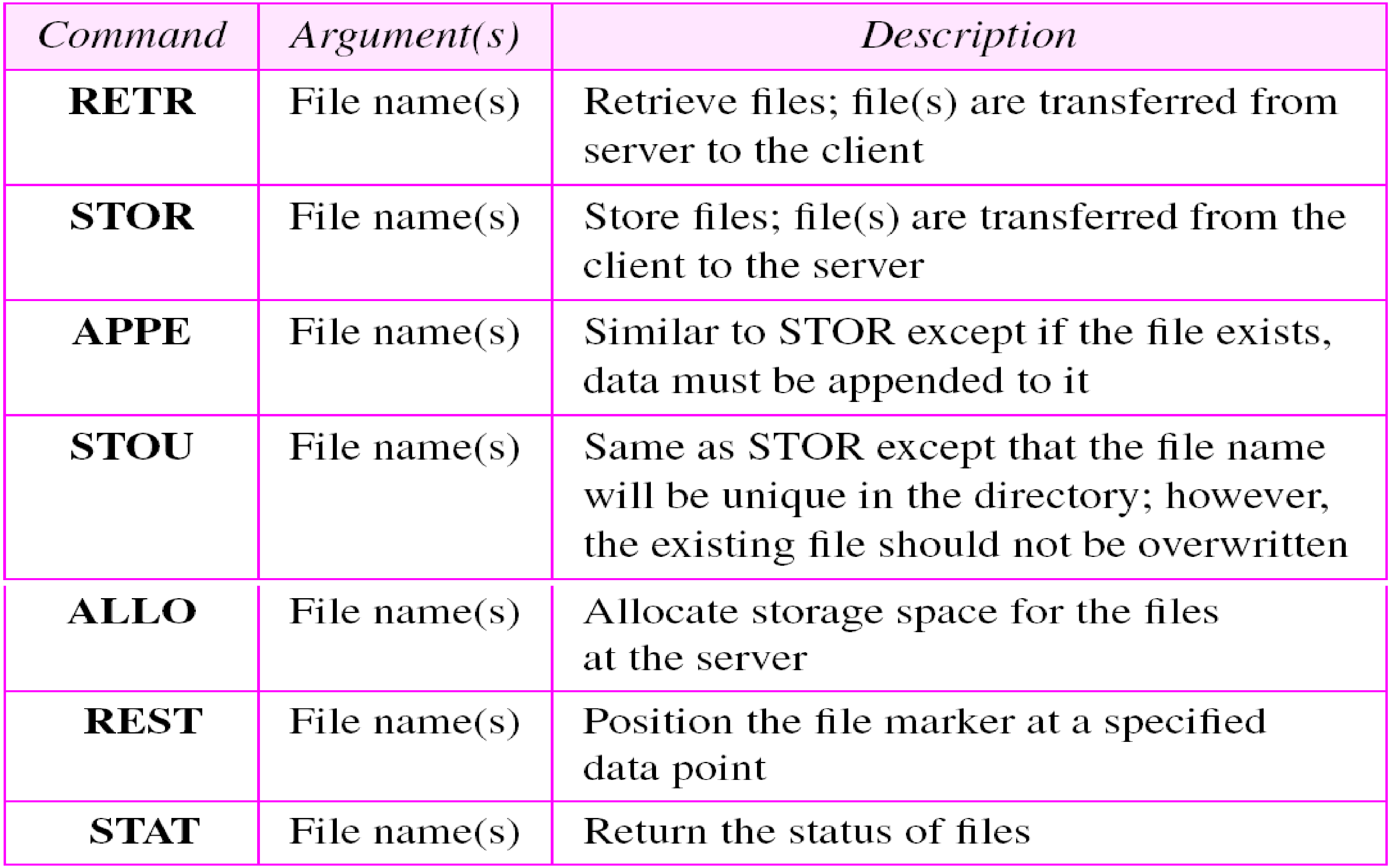
Command	*Argument(s)*	*Description*
TYPE	A (ASCII), E (EBCDIC), I (Image), N (Nonprint), or T (TELNET)	Define the file type and if necessary the print format
STRU	F (File), R (Record), or P (Page)	Define the organization of the data
MODE	S (Stream), B (Block), or C (Compressed)	Define the transmission mode

FTP Port Defining Commands



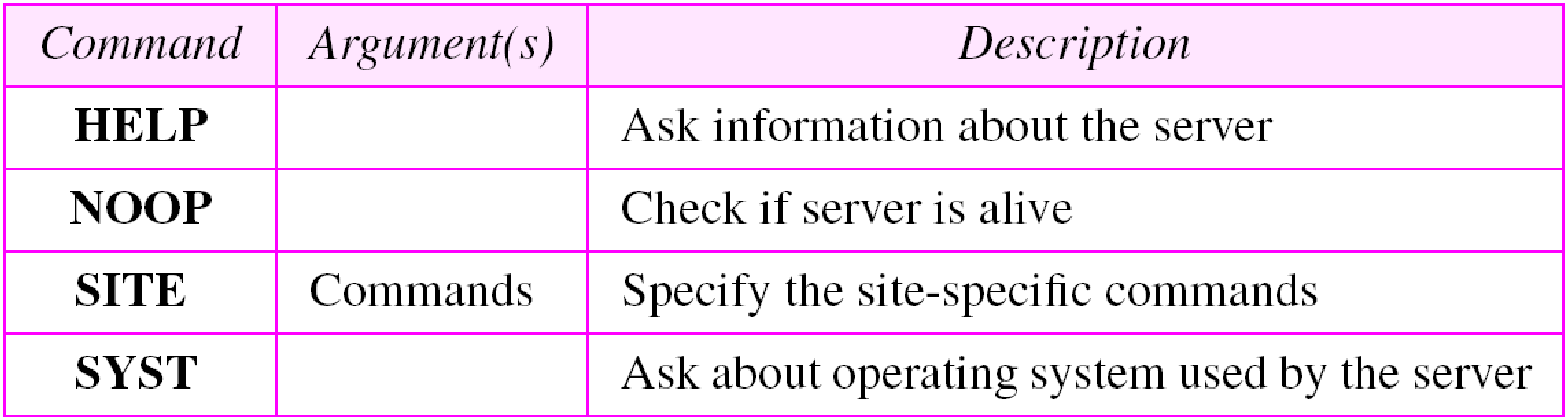
Command	*Argument(s)*	*Description*
PORT	6-digit identifier	Client chooses a port
PASV		Server chooses a port

FTP File Transfer Commands



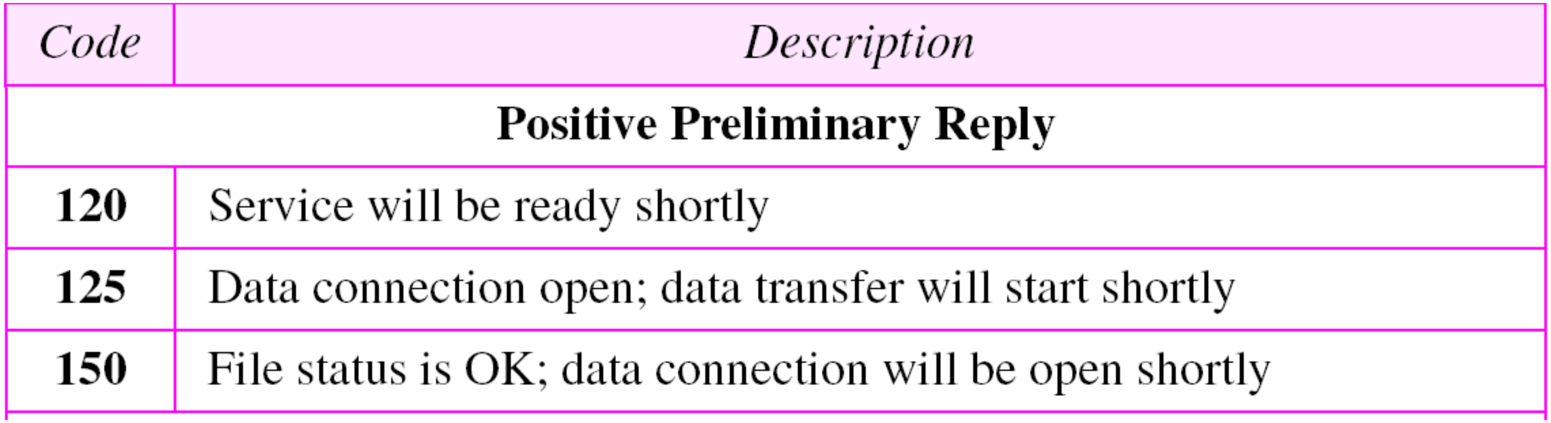
Command	*Argument(s)*	*Description*
RETR	File name(s)	Retrieve files; file(s) are transferred from server to the client
STOR	File name(s)	Store files; file(s) are transferred from the client to the server
APPE	File name(s)	Similar to STOR except if the file exists, data must be appended to it
STOU	File name(s)	Same as STOR except that the file name will be unique in the directory; however, the existing file should not be overwritten
ALLO	File name(s)	Allocate storage space for the files at the server
REST	File name(s)	Position the file marker at a specified data point
STAT	File name(s)	Return the status of files

FTP Miscellaneous Commands



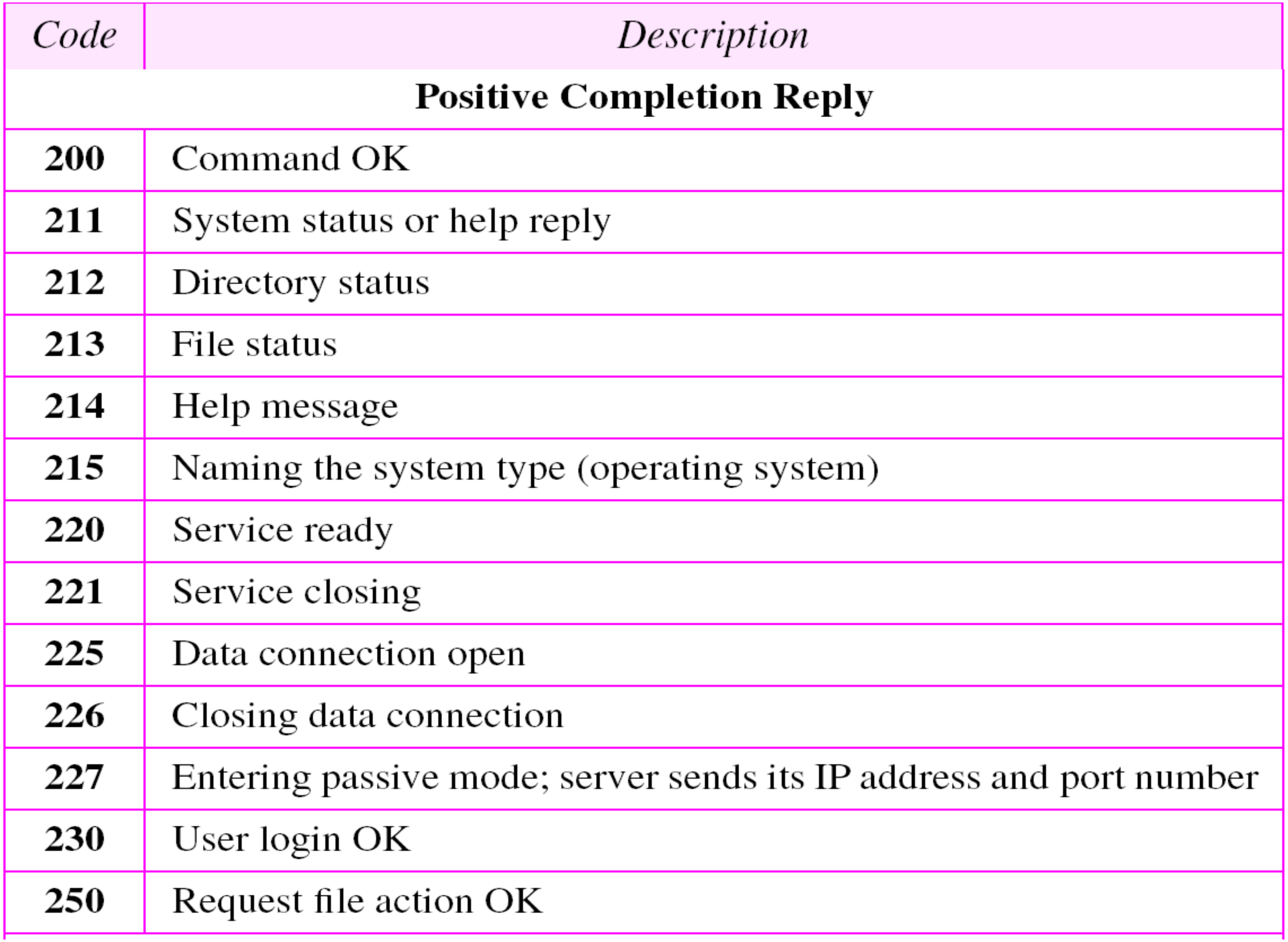
Command	*Argument(s)*	*Description*
HELP		Ask information about the server
NOOP		Check if server is alive
SITE	Commands	Specify the site-specific commands
SYST		Ask about operating system used by the server

FTP Responses



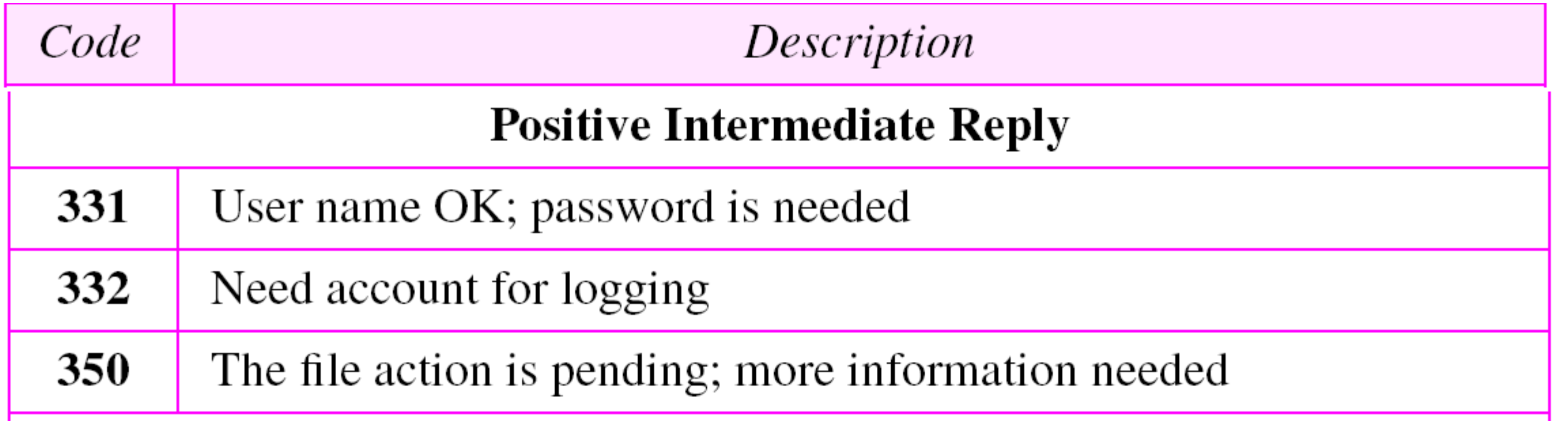
Code	*Description*
Positive Preliminary Reply	
120	Service will be ready shortly
125	Data connection open; data transfer will start shortly
150	File status is OK; data connection will be open shortly

FTP Responses



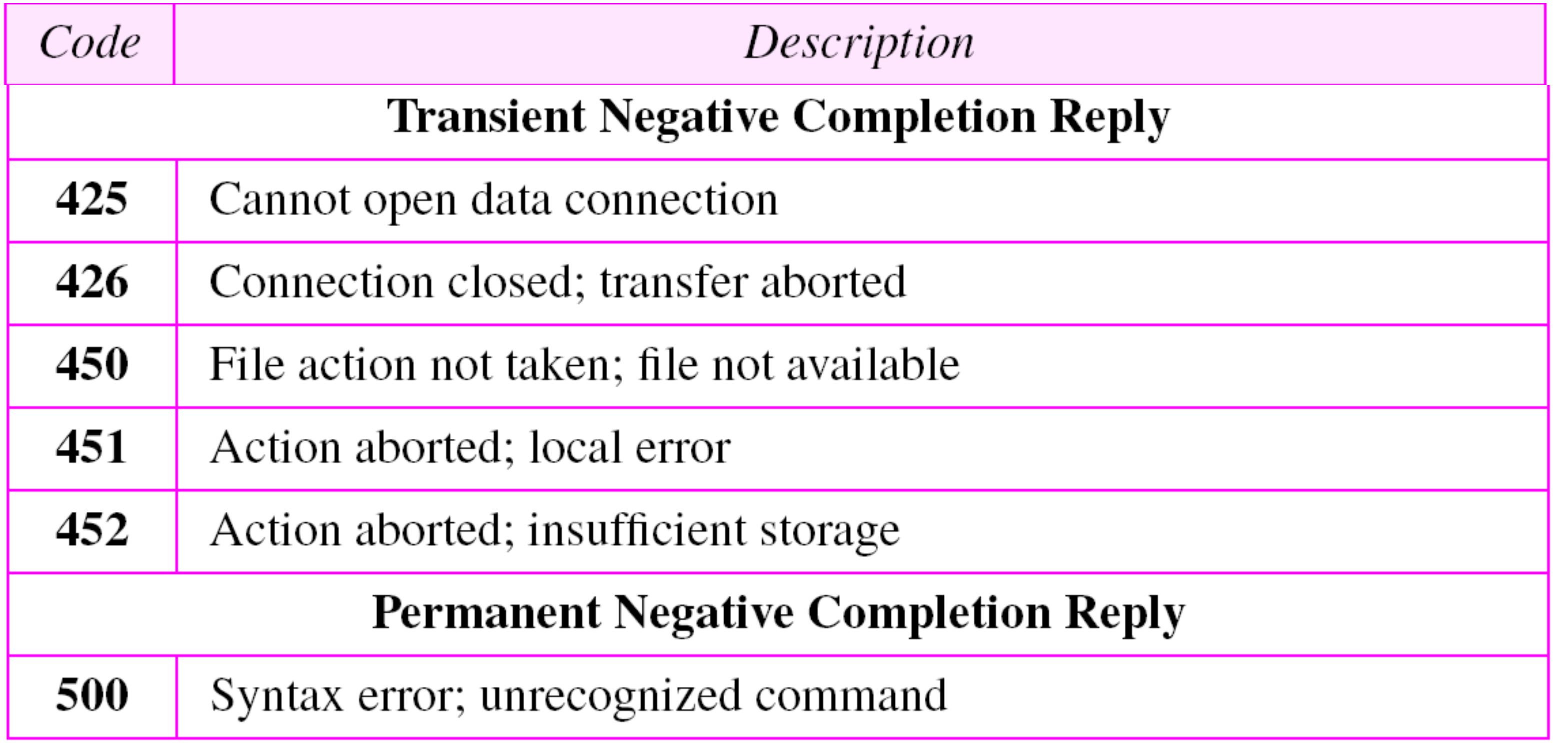
Code	*Description*
Positive Completion Reply	
200	Command OK
211	System status or help reply
212	Directory status
213	File status
214	Help message
215	Naming the system type (operating system)
220	Service ready
221	Service closing
225	Data connection open
226	Closing data connection
227	Entering passive mode; server sends its IP address and port number
230	User login OK
250	Request file action OK

FTP Responses



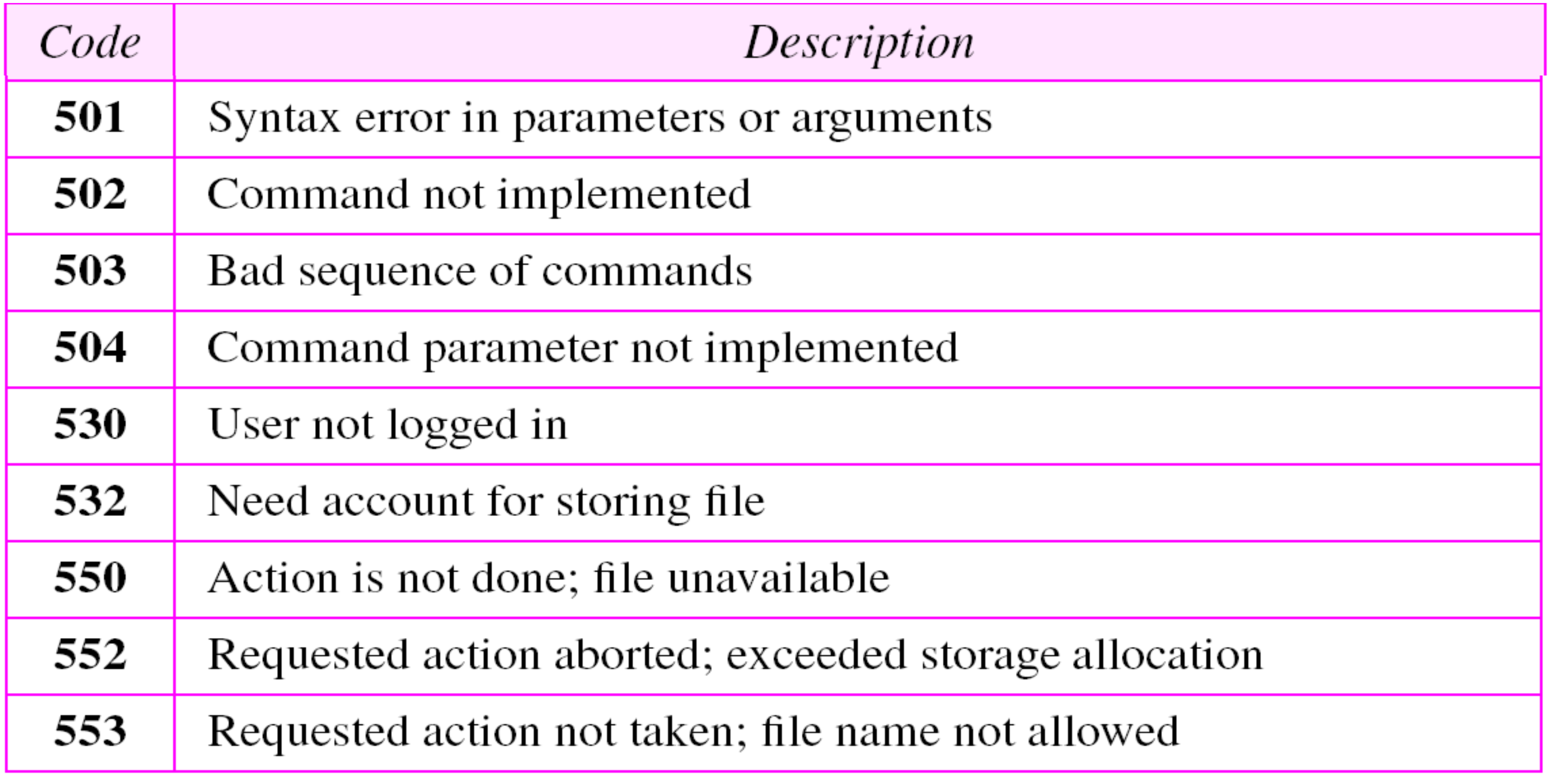
Code	*Description*
Positive Intermediate Reply	
331	User name OK; password is needed
332	Need account for logging
350	The file action is pending; more information needed

FTP Responses



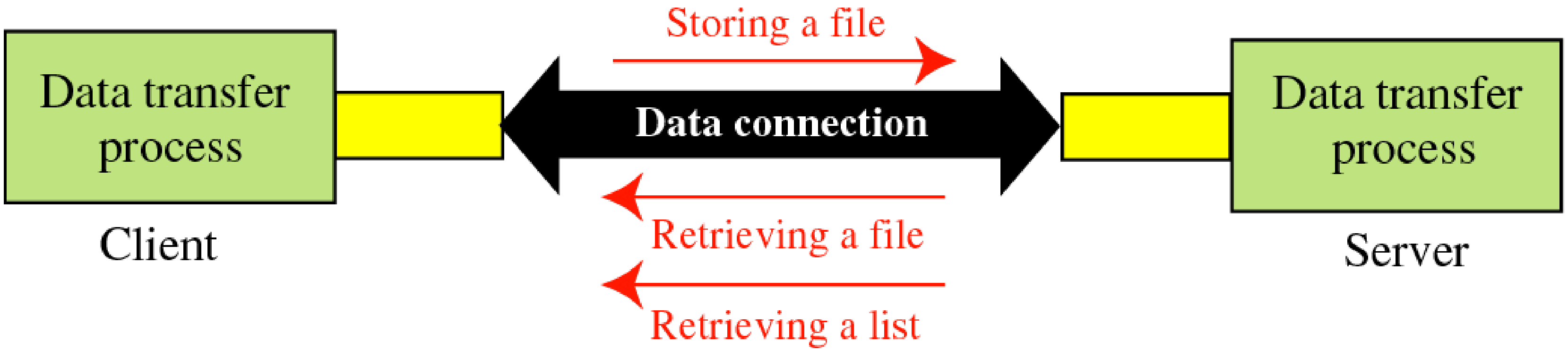
Code	*Description*
Transient Negative Completion Reply	
425	Cannot open data connection
426	Connection closed; transfer aborted
450	File action not taken; file not available
451	Action aborted; local error
452	Action aborted; insufficient storage
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command

FTP Responses

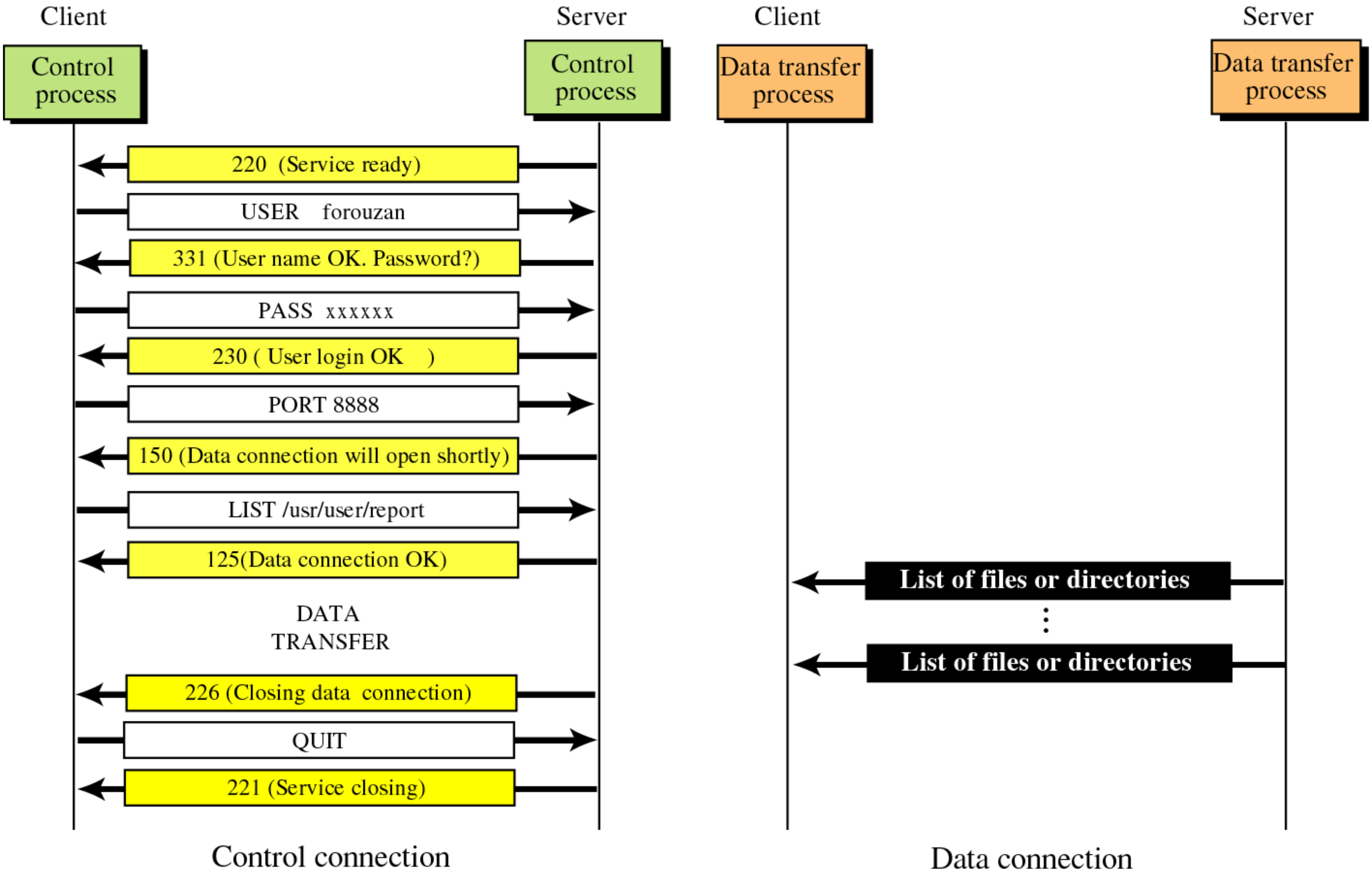


Code	*Description*
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented
530	User not logged in
532	Need account for storing file
550	Action is not done; file unavailable
552	Requested action aborted; exceeded storage allocation
553	Requested action not taken; file name not allowed

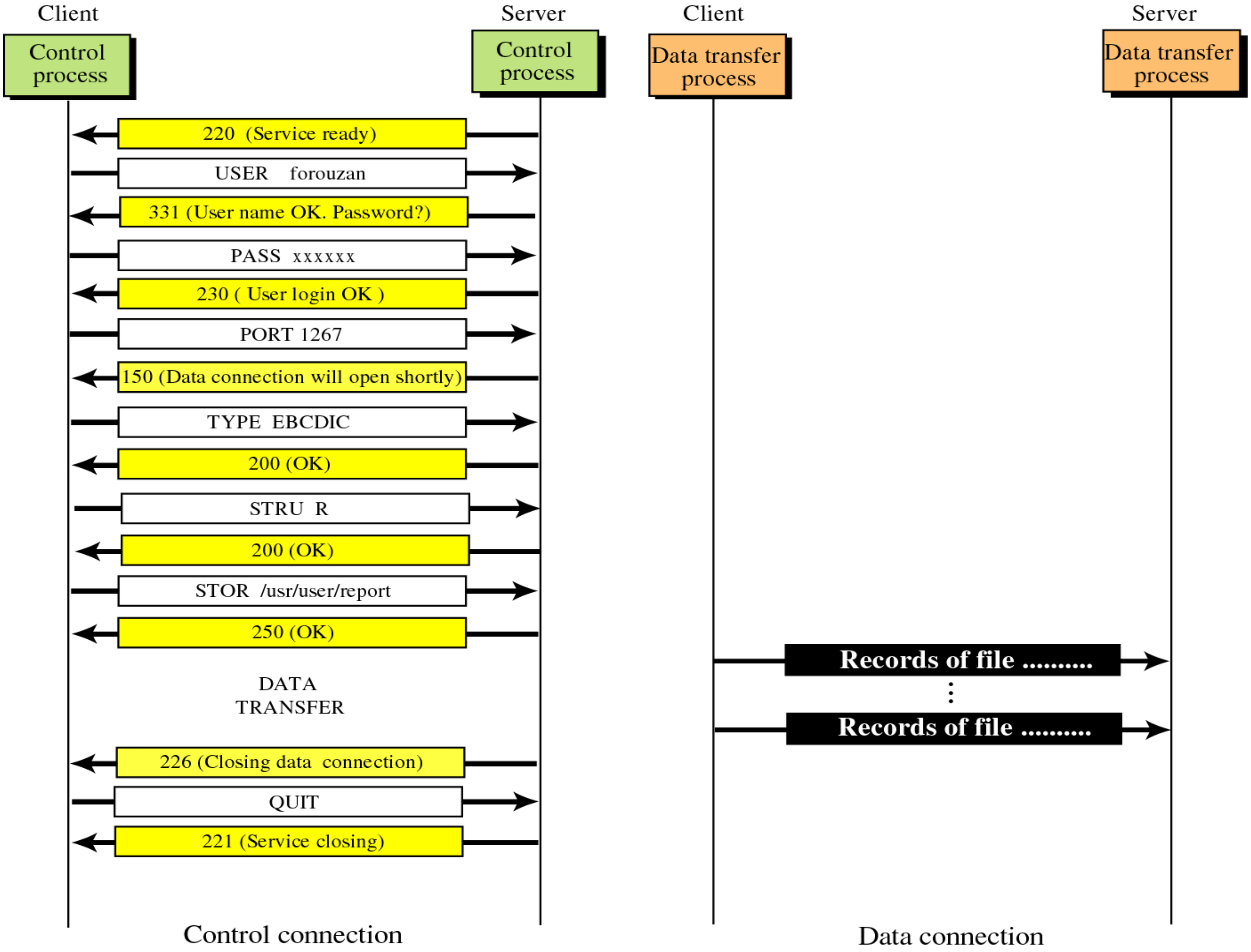
FTP File transfer



FTP: Example 1



FTP: Example 2



Client

Control process

Server

Control process

Client

Data transfer process

Server

Data transfer process

220 (Service ready)

USER forouzan

331 (User name OK. Password?)

PASS xxxxxx

230 (User login OK)

PORT 1267

150 (Data connection will open shortly)

TYPE EBCDIC

200 (OK)

STRU R

200 (OK)

STOR /usr/user/report

250 (OK)

DATA TRANSFER

Records of file ....

⋮

Records of file ....

226 (Closing data connection)

QUIT

221 (Service closing)

Control connection

Data connection

Summary

- DNS
  - Name Space
  - Resource Record
  - DNS Server
- HTTP
  - URL
  - HTML
  - HTTP Methods
  - HTTP Headers
- FTP
  - Control and Data Connections
  - Commands and Replies

Next

Network Security

- Secrecy, Authentication, Nonrepudiation, and Integrity
- Cryptography
    - Plain and Cipher Texts
    - Substitution Cipher
    - Transposition Cipher
    - Product Cipher
- Public Key Algorithm
- Digital Signature