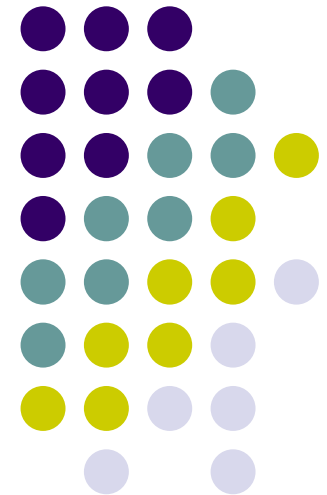


Lecture 9: Application layer

Reading Chapter 7
Computer networks, Tanenbaum

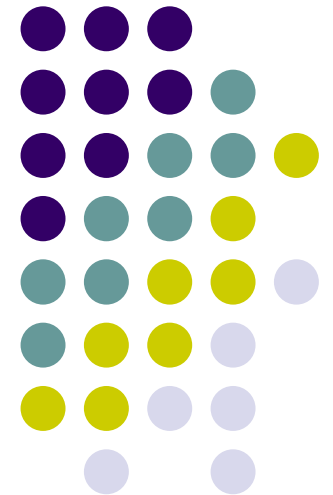




Contents

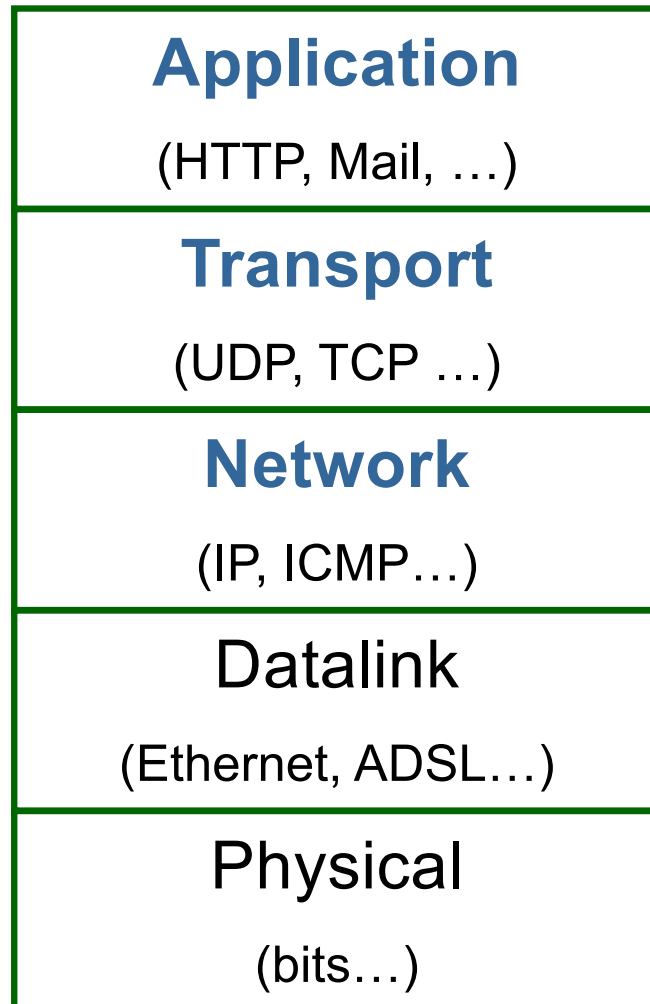
- Application layer
 - Fundamental concepts
 - Case study: HTTP, Mail, FTP...

Fundamental concepts





Application layer in OSI model



**Protocols communication
between parties of the
application**

Transmission data between application

Application and service?



VoIP

MUSIC ONLINE

GAME ON LINE

CHAT

VoD

SMS

e-Office

e-BANK

MAIL

E-learning

WEB

YOUTUBE

VIDEO CONFERENCE

FTP

EBAY

GOOGLE

SKYPE

Social networks

SSH

NEWS

BITTORENT

E-COMMERCE

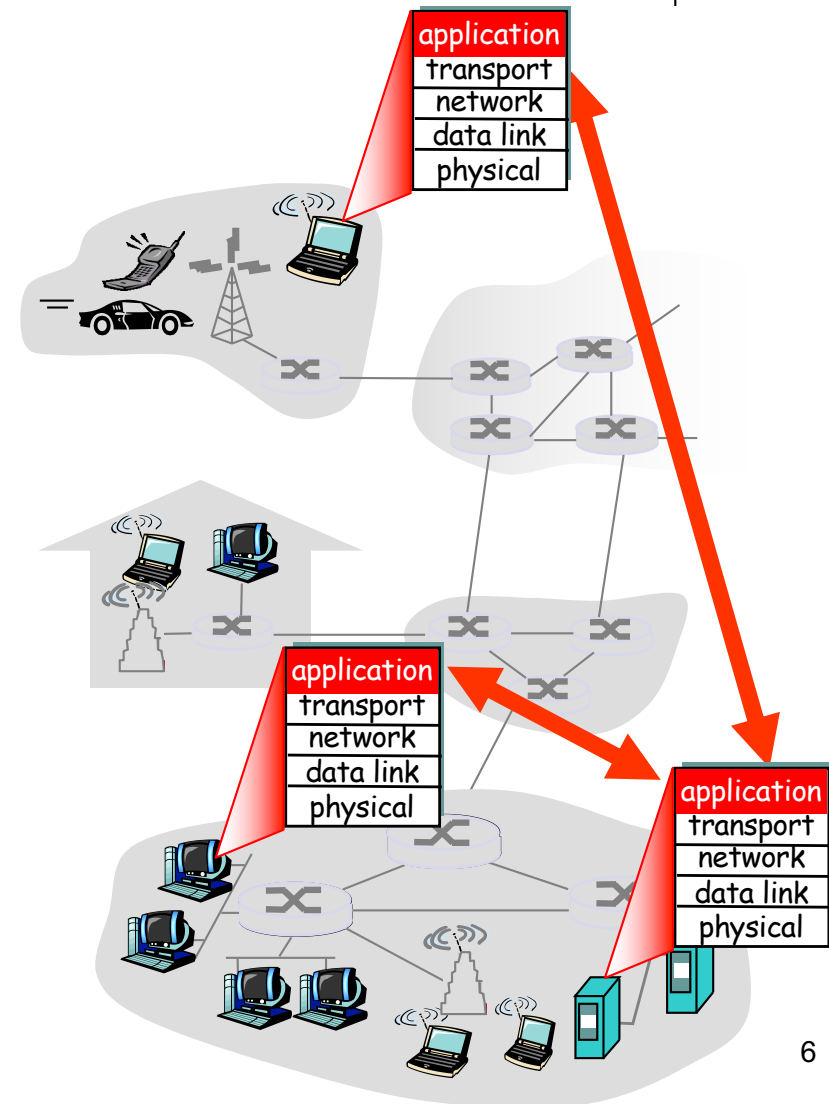
GRID

e-Government

Application and application protocol



- Application protocol
 - Define communication rule
 - Use service of transport layer (TCP/UDP...)
- Application:
 - Is a process on the internet. They communicate to each other by exchanging messages.
 - Runs on end systems
 - Use application protocol for providing service
- Example of application/protocol:
 - Web (HTTP)
 - Mail (SMTP/POP/IMAP) ...

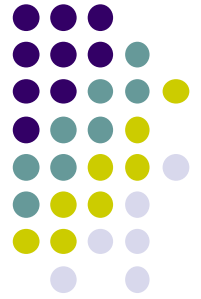




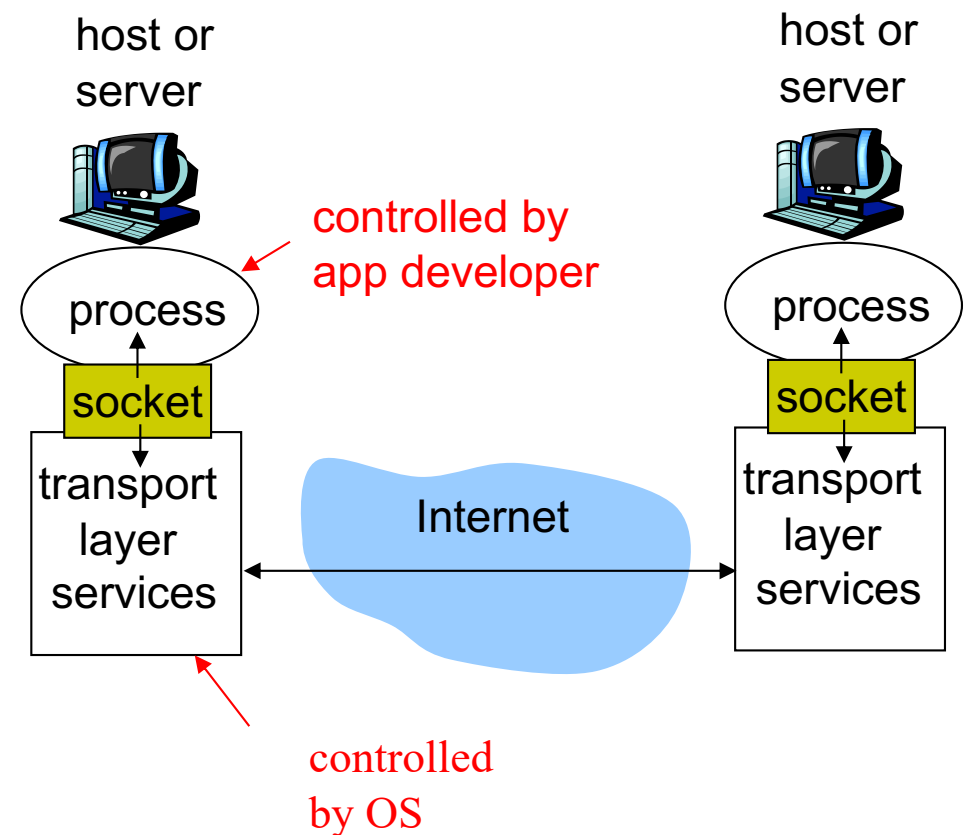
Components of an application

- **Application software is compose of**
 - **User interface:**
 - Interfacing with users,
 - e.g. Web browser (Firefox, IE), mail reader(Thunderbird, Outlook,...)
 - Implement one part of application protocol
 - **Server program:**
 - Cung cấp dịch vụ cho người sử dụng
- **Application process:** the application software running on an OS

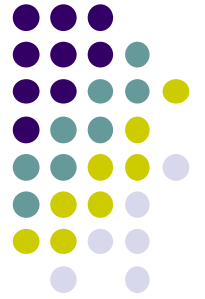
Communication between process on the Internet



- Socket is an interface between an application process and transport layer
- Socket is defined by
 - Port
 - IP address
 - Transport protocol (TCP or UDP)
- Socket API (Application Programming Interface): Allow application to choose parameters for transport service
 - Choose transport protocol
 - Type of IO communication ...

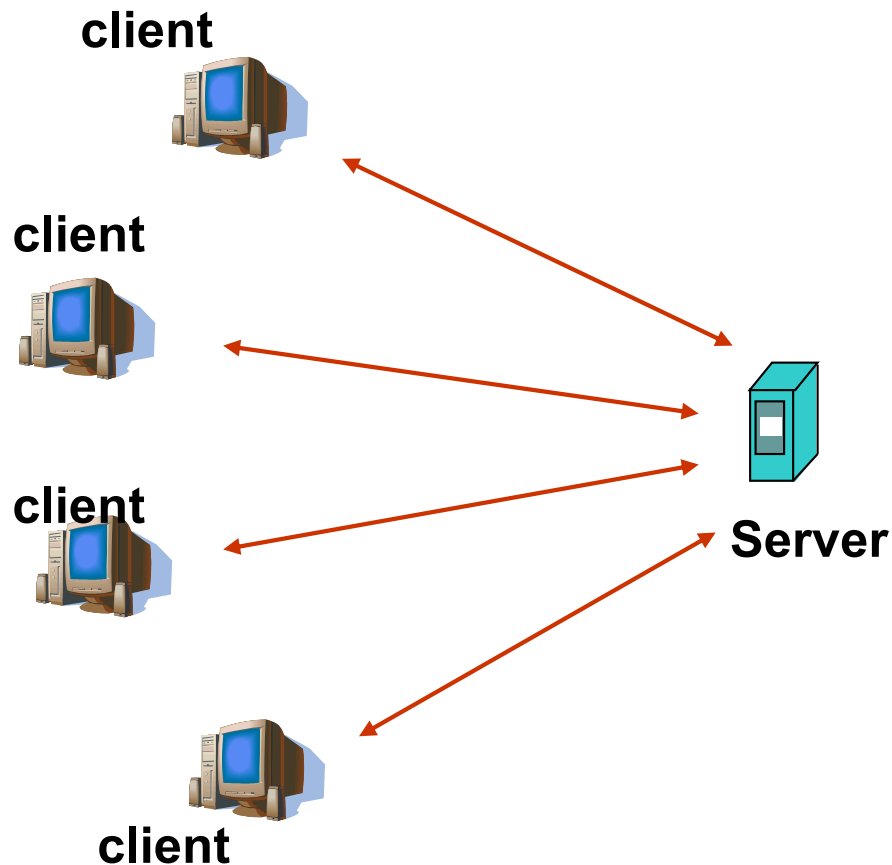


Application architecture

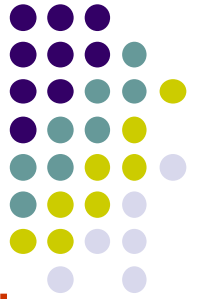


- Client-server
- P2P
- Hybrid

Client-server

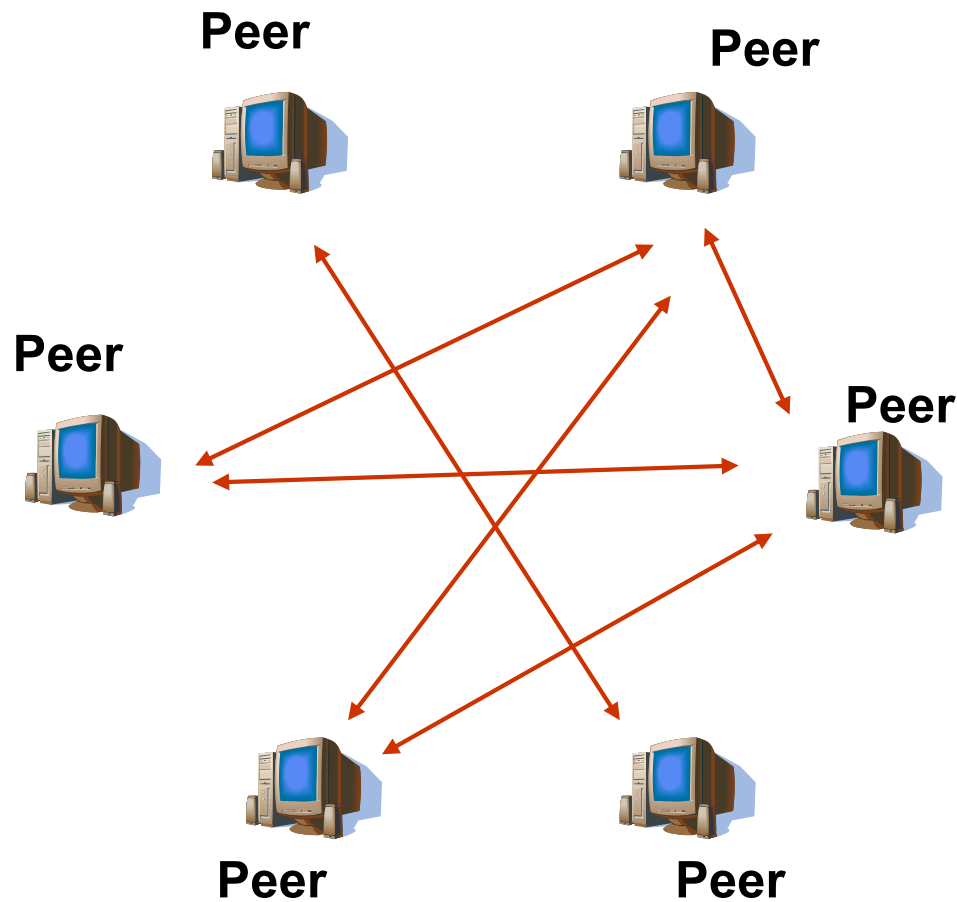


- Two kind of components: client and server
- **Client**
 - Client sends requests for service to server
 - Clients do not contact directly to each other
- **Server**
 - Always online waiting for service requests from clients
 - There may be backup servers for assuring high availability in failures
- e.g. Web, Mail, ...



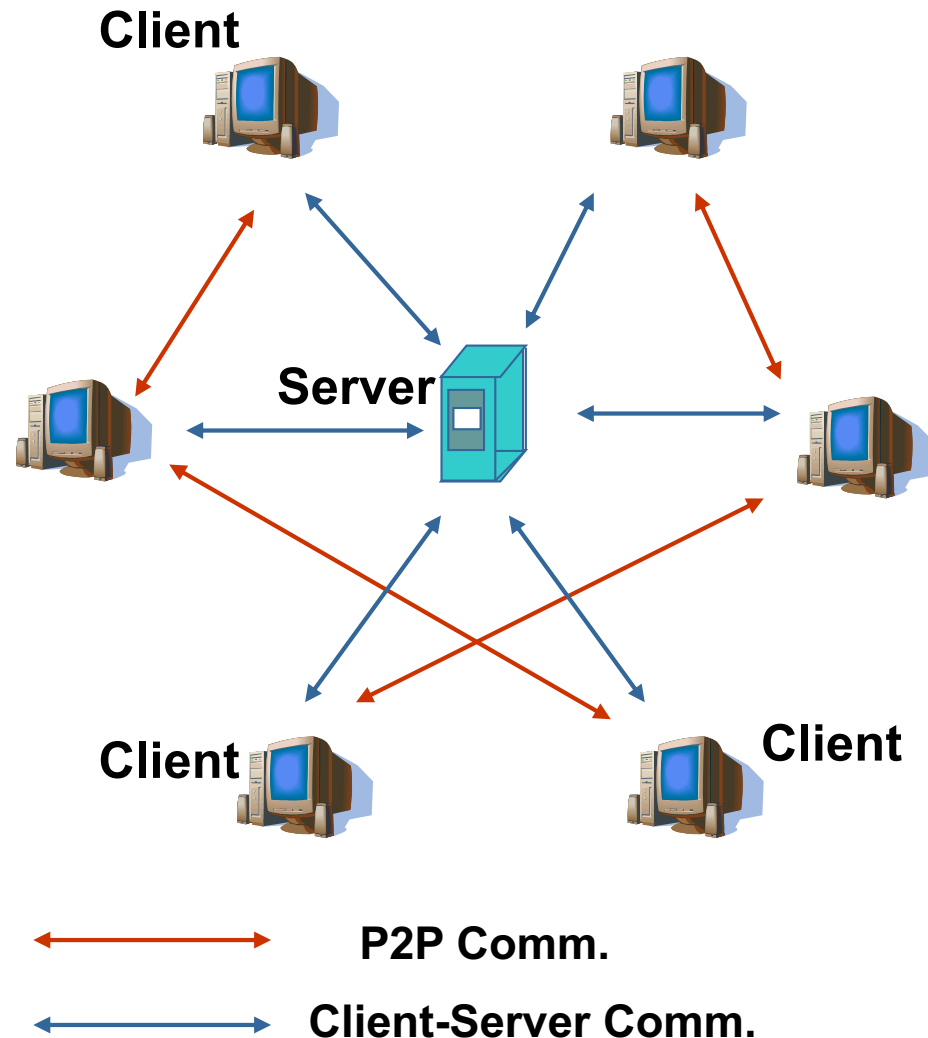


Pure Peer-to-peer architecture



- No center server, only peers as components
- Peers have equal role in the system
- Any two peers can communicate directly to each other but only when both are online.
- Peer does not need to be online all the time
- E.g. Gnutella, Bittorrent

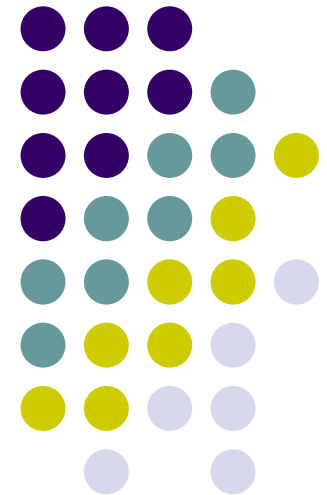
Hybrid architecture



- A center server for user management, indexing for search purpose.
- Clients communicate directly to each other after authentication process with server.
- E.g. Skype (before 2016)
 - Skype server manage user lists, authentication
 - After authentication users communicate directly to each other

Case study 1: HTTP and WWW

Reading 7.3
Computer Networks, Tanenbaum

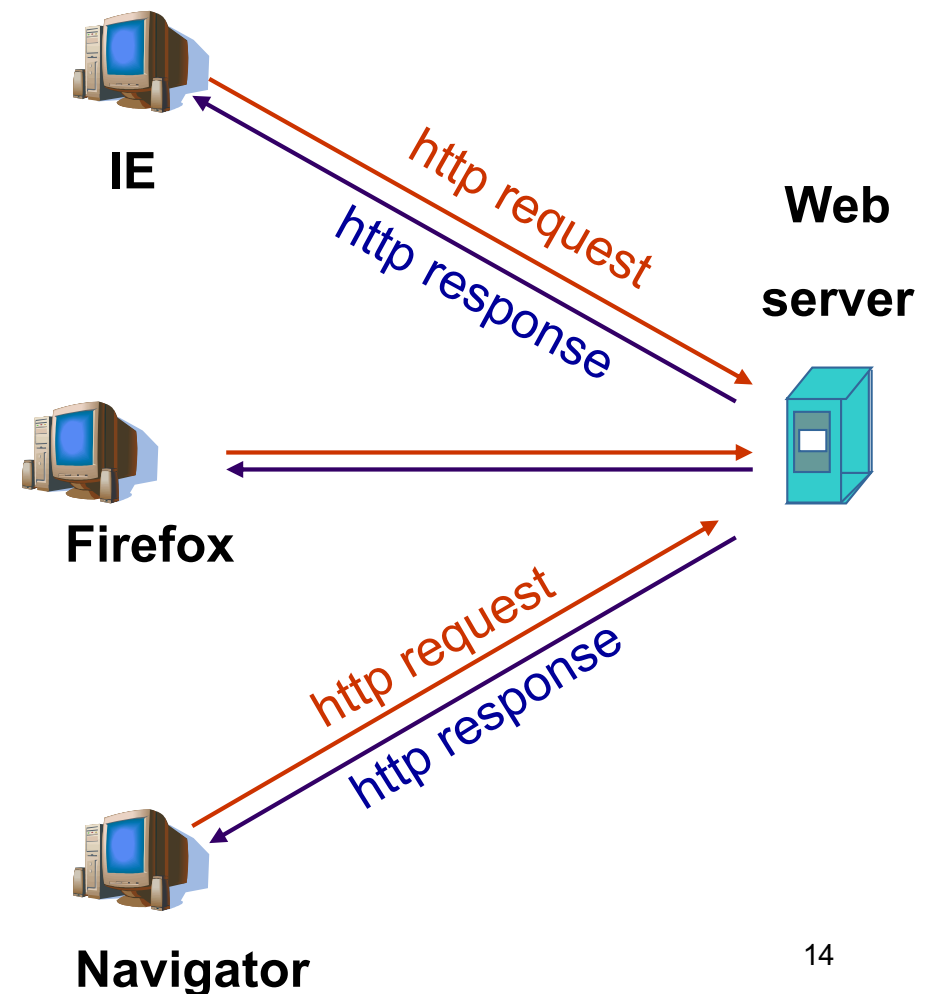




HTTP and Web

- WWW: World Wide Web
 - Application for exchanging the HTML documents (HyperText Markup Language) over Internet
 - WWW use HTTP protocol
- HTTP: HyperText Transfer Protocol
 - Client/Server model
 - Client (Web browser) requests for webpages and displays them on its interface
 - Server: Receive request from client and return results under the form of webpage.

Web clients





How HTTP works?

- Server open a TCP socket by default at port 80 waiting for
- Client initiates a TCP connection to server
- Server accept the connection request
- Exchange HTTP message
 - HTTP Request
 - HTTP Response
- Close connection TCP



Format of HTTP request

- ASCII encoding (readable using text editor)

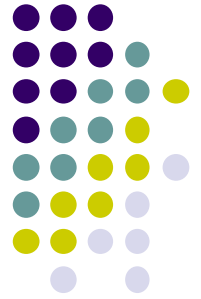
request line
(GET, POST,
HEAD commands)

header
lines

CR, LF
(extra carriage return, line feed)

indicates end
of message

```
GET /dccn/index.html HTTP/1.1
Host: www.it-hut.edu.vn
User-agent: Mozilla/4.0
Connection: close
Accept-language: en-us
```

Format of HTTP response

status line
(protocol
status code
status phrase)

header
lines

data, e.g.,
requested
HTML file

```
HTTP/1.1 200 OK
Connection close
Date: Tue, 16 Mar 2008 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 15 Mar 2008 .....
Content-Length: 8990
Content-Type: text/html

data data data data data ...
```



Types of HTTP connections

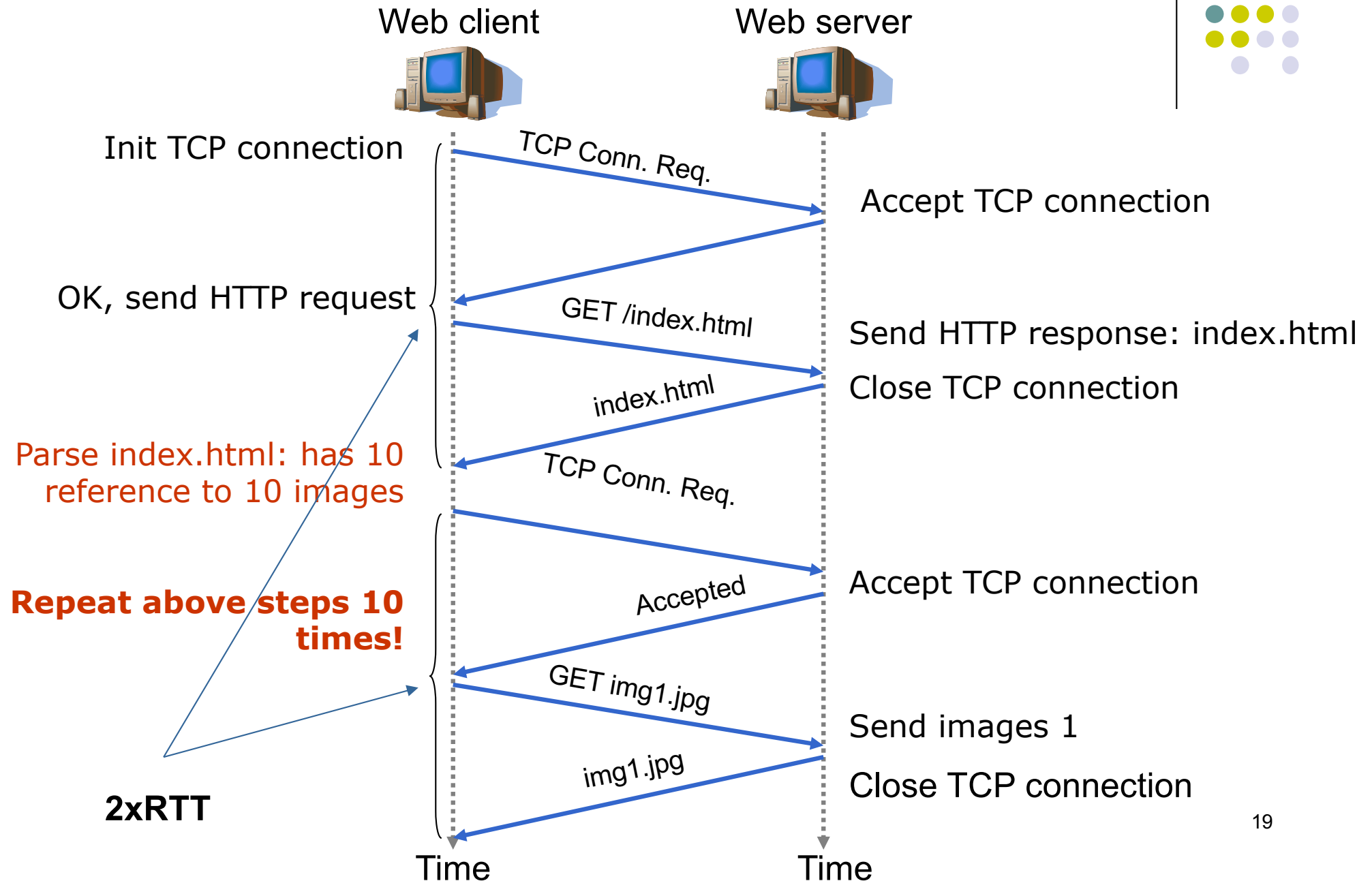
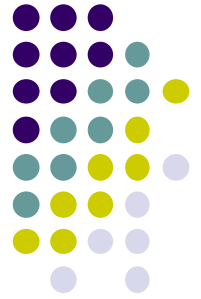
HTTP non-persistent

- Only one web object (text or image) is transferred over a connection TCP
- Option by default in HTTP/1.0
- HTTP 1.0: RFC 1945

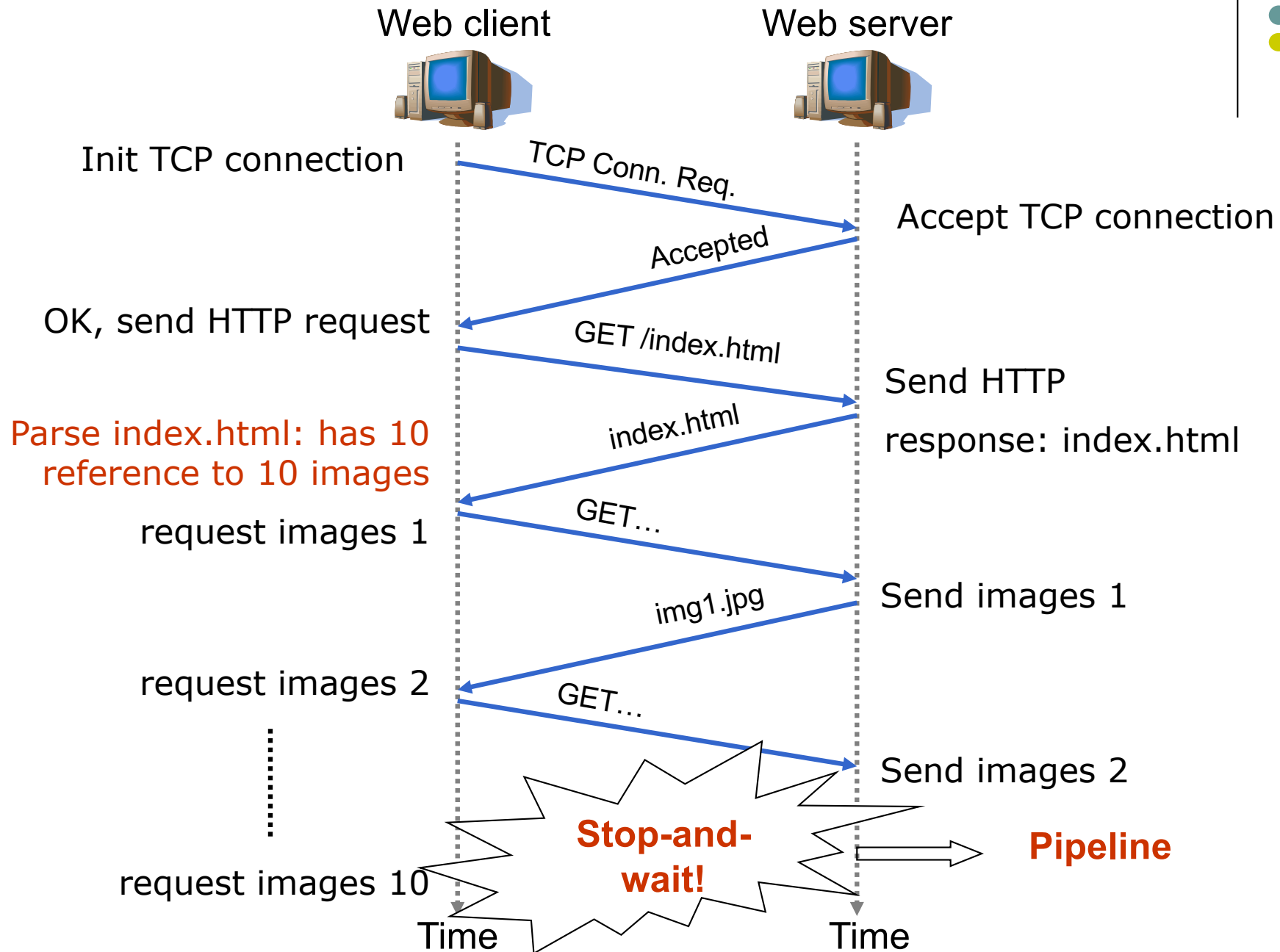
HTTP persistent

- Many web objects can be sent over a connection TCP.
- Option by default in HTTP/1.1
- HTTP 1.1: RFC 2068

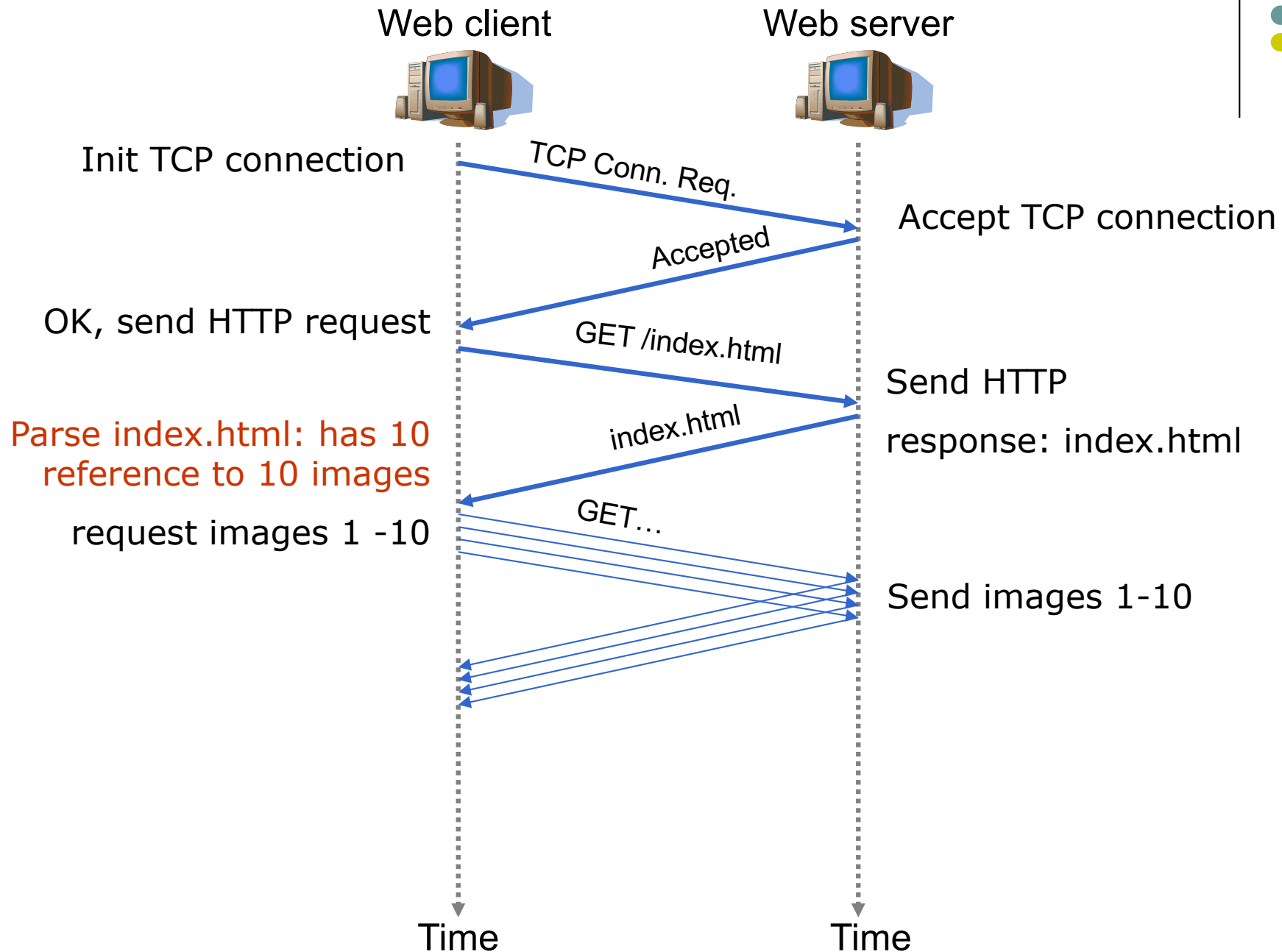
Operation of HTTP/1.0



Operation of HTTP/1.1



HTTP/1.1 with pipeline





Methods in HTTP request message

HTTP/1.0

- GET: get an webpage
- POST: submitting a form
- HEAD: ask for the header of an webpage

HTTP/1.1

- GET, POST, HEAD
- PUT
 - Upload an webpage to the server under address given in URI, file content is in the body of the message
- DELETE
 - Delete a file given in the URI

Attention: Even with GET, user can sends parameters to servers in URL. Ex:

<http://www.google.com/search?q=computer+network&flags=68&num=10>



Status code in Response message

Status code is in the first line of the Response message

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

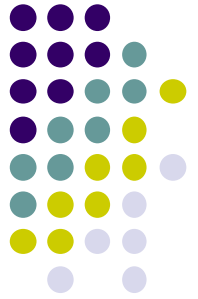
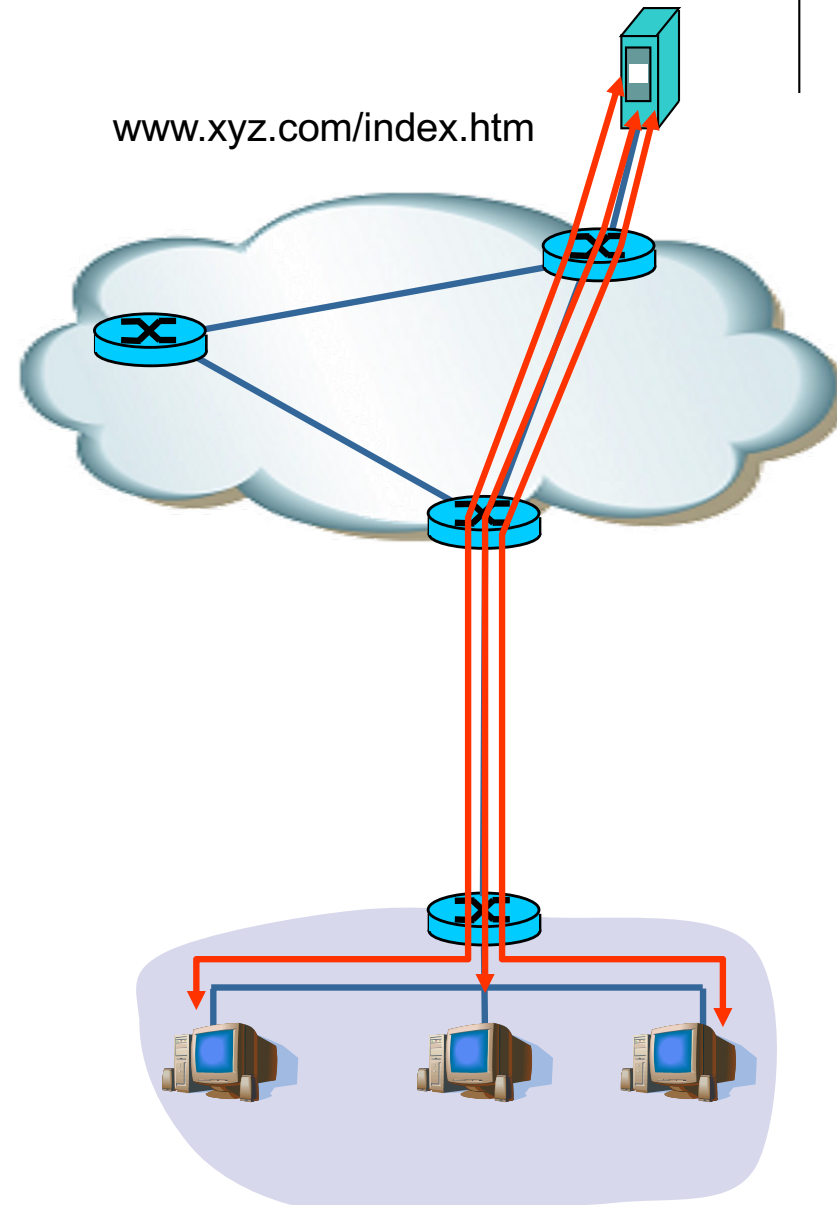
404 Not Found

- requested document not found on this server

505 HTTP Version Not Supported

Web cache

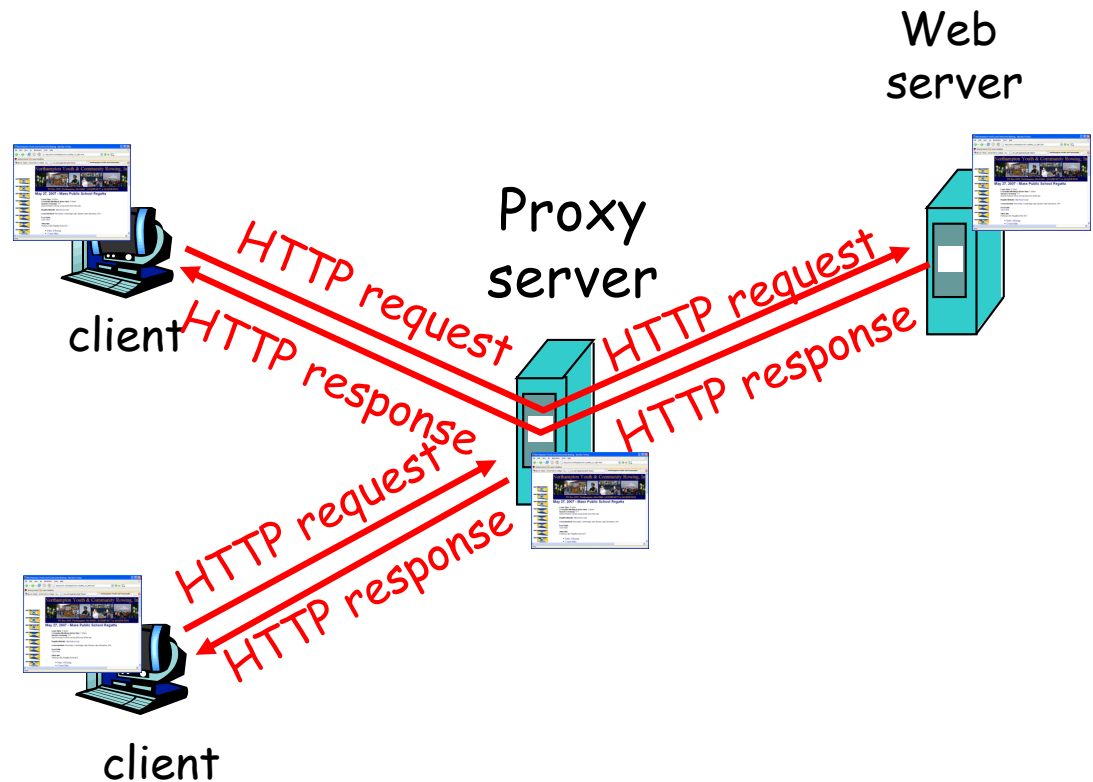
- “Cache”: buffer memory
- Similar notion of computer cache
 - L1 cache, L2 cache
 - “cache miss”, “cache hit”
- Case study:
 - An organization has uniquely an Internet line.
 - Multiple users can access to the same webpages, ex: news pages
 - Solution for improving the performance?



Web cache - proxy



- Users connects to web server through a web proxy.
- Web browsers send a request to a web page to the proxy
 - Miss: The required web page is not found Proxy gửi yêu cầu tới máy chủ web, trả lời trình duyệt và lưu đệm đối tượng web
 - Hit: Proxy trả đối tượng web cho trình duyệt





Web caches

- Proxy: Vừa là client, vừa là server
- Sử dụng bởi các ISP nhỏ, các tổ chức như trường học, công ty...
- Ảnh hưởng của proxy
 - Làm giảm lưu lượng web trên đường ra Internet
 - Có thể làm giảm thời gian đáp ứng
 - Thử phân tích vài trường hợp
 - cache hit
 - cache miss
 - proxy bị quá tải
 - Trang web thay đổi/trang web động?



Local cache

- Web pages could be stored in local server (local cache)
- Using local cache for
 - Reading web offline
 - Improve performance in accessing web pages



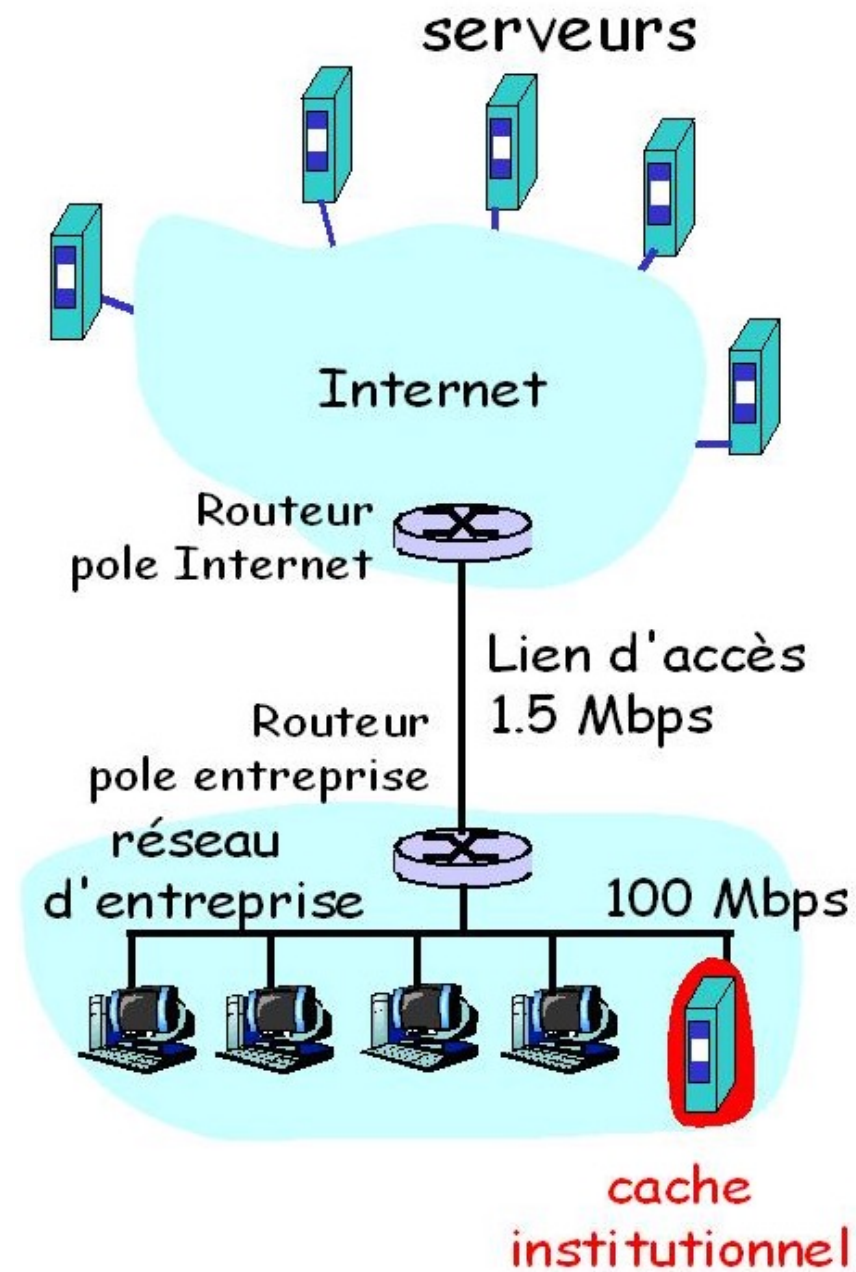
Exercise

- A company having an Internet connection from their LAN with data rate 1.5 Mbps.
- Webpage to be downloaded from the Internet has the average size 900 kbits
- The frequency of accessing files on the web is $=1,5$ times/s.
- Duration for sending an HTTP request from company router to the Internet is 2 (s)
- What is the average responding time in following cases?

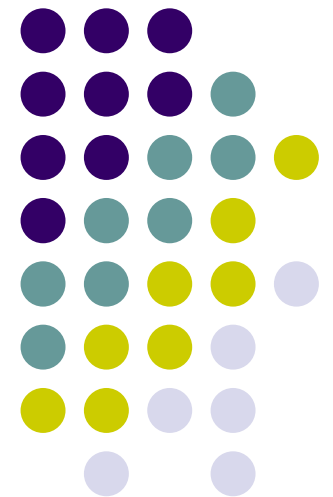


Exercise (cont.)

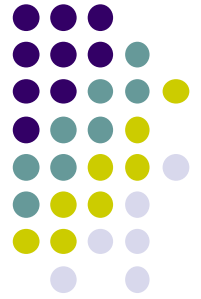
- If all the links have already some load.
Transmission time on a loaded link is calculated by the following formular:
 - $= W_m / (1 - \rho)$,
 - W_m : Transmission time without load
 - ρ : load (ratio of traffic/capacity)
- If there is a cache
 - Assume that ccache hit rate: $p=0.4$.
 - Hint:
 - Use the formular: $T_m = \sum p_i * T_i$,
 - p_i probability having responding time T_i



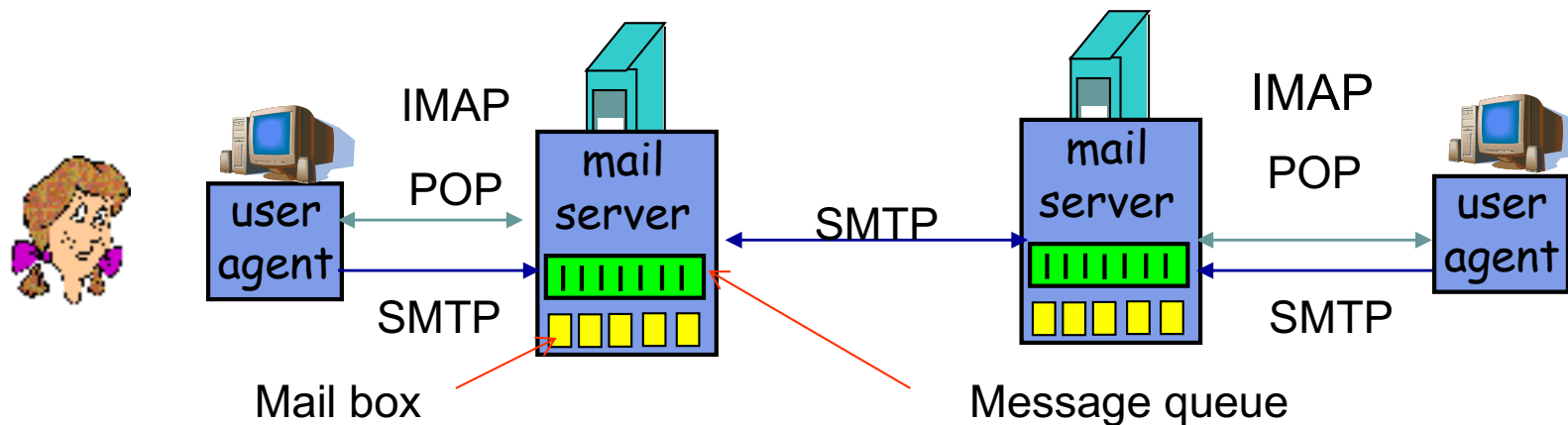
Case study 2: Email

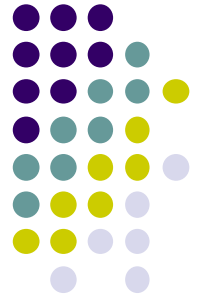


Thư điện tử



- MUA (Mail User Agent)
 - Lấy thư từ máy chủ, gửi thư đến máy chủ
 - e.g. Outlook, Thunderbird...
- MTA (Mail Transfer Agent): :
 - Chứa hộp thư đến của NSD (mail box)
 - Hàng đợi để gửi thư đi
 - e.g. Sendmail, MS Exchange...
- Giao thức:
 - Chuyển thư: SMTP-Simple Mail Transfer Protocol
 - nhận thư
 - POP – Post Office Protocol
 - IMAP – Internet Mail Access Protocol

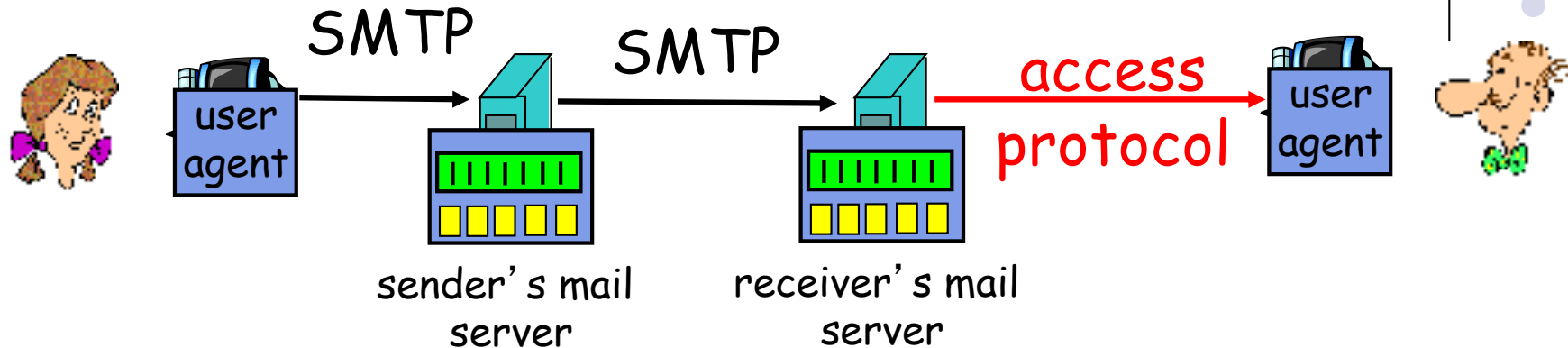




Giao thức SMTP

- RFC 2821
- TCP, port 25: Chuyển thư từ client đến server và giữa các server với nhau
- Tương tác yêu cầu/trả lời
 - Yêu cầu: Lệnh với mã ASCII
 - Trả lời: mã trạng thái và dữ liệu

Các giao thức nhận thư



- POP: Post Office Protocol [RFC 1939]
 - Đăng nhập và lấy hết thư về
- IMAP: Internet Mail Access Protocol [RFC 1730]
 - Phức tạp hơn POP
 - Cho phép lưu trữ và xử lý thư trên máy chủ
 - Quota?



Web Mail

- Sử dụng Web browser như một MUA
- MUA và MTA giao tiếp thông qua HTTP
- Mails được lưu trữ trên máy chủ
- E.g.
 - Gmail,
 - Hotmail,
 - Yahoo! Mail, etc.
- Ngày nay, rất nhiều các MTA cho phép truy cập thông qua giao diện web
 - <http://mail.hut.edu.vn>
 - <http://mail.fit.hut.edu.vn>

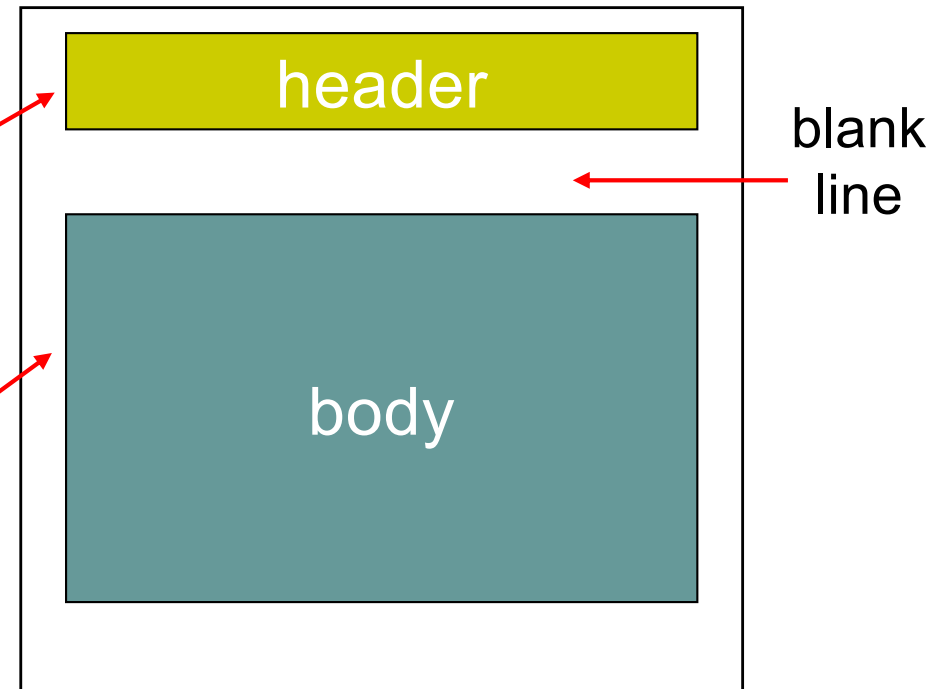
Khuôn dạng thông điệp thư điện tử



SMTP: Giao thức để truyền thư

RFC 822: Định nghĩa khuôn dạng

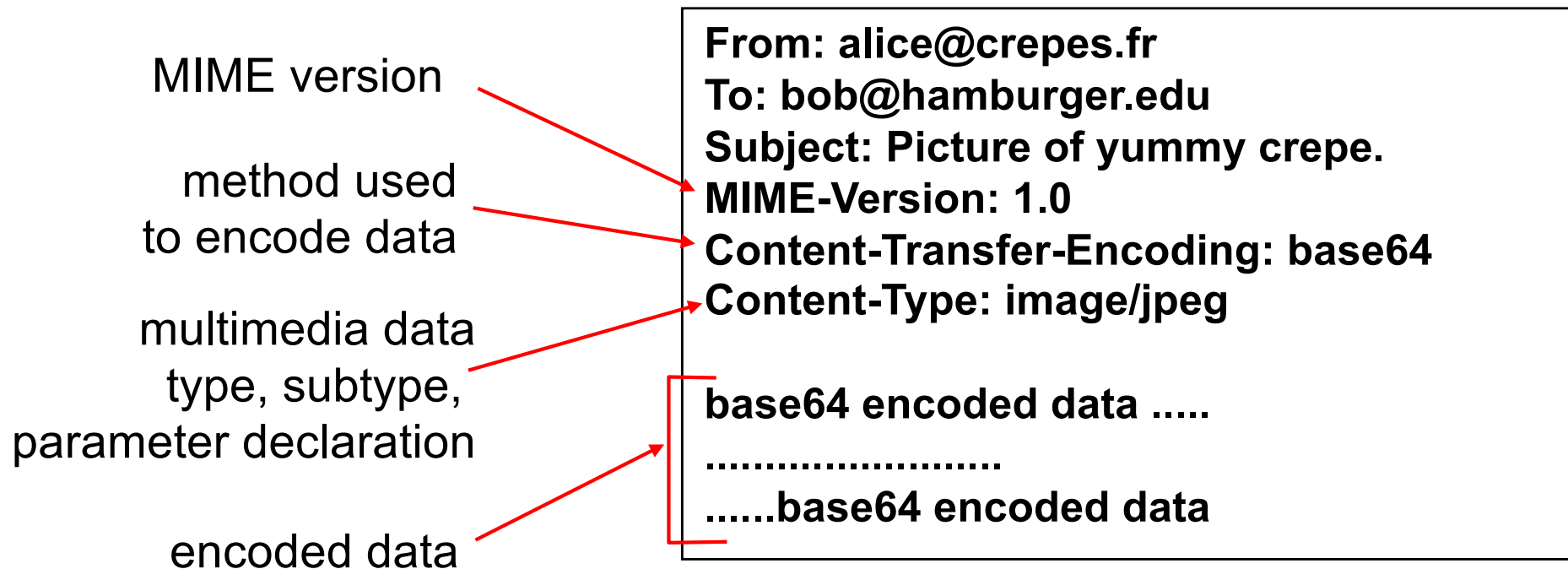
- Phần đầu
 - To:
 - From:
 - Subject:
- Phần thân
 - mã hóa dưới dạng mã ASCII



Đề chuyển dữ liệu đa phương tiện: multimedia extensions

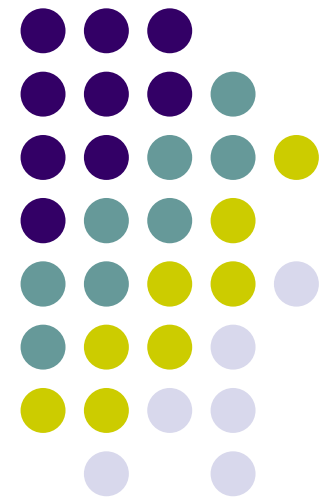


- MIME: multimedia mail extension, RFC 2045, 2056
- Thêm một dòng trong phần đầu chỉ rõ khuôn dạng dữ liệu gửi đi

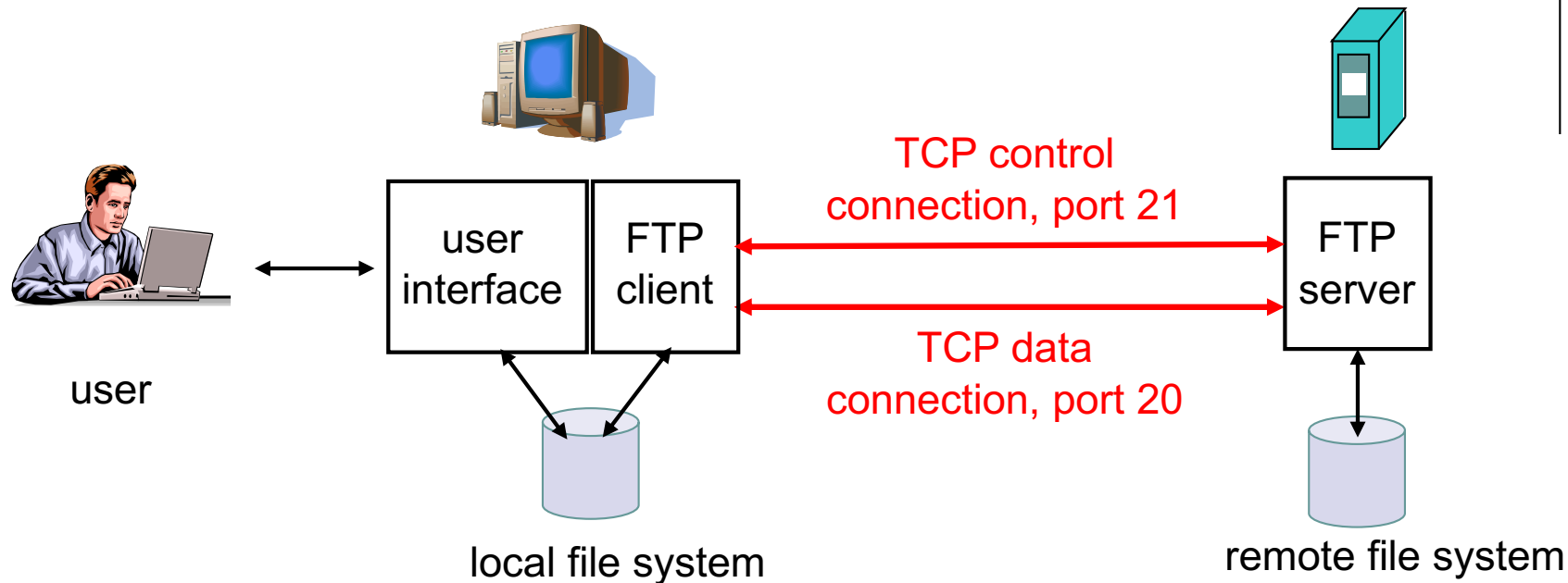
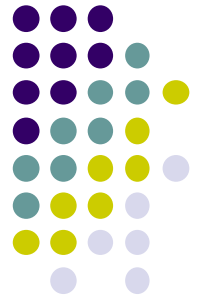


Case Study 3

Ứng dụng truyền tệp



FTP: File Transfer Protocol



- Mô hình Client-server
- Trao đổi file giữa các máy
- RFC 959
- Sử dụng TCP, cổng 20, 21
- Điều khiển **Out-of-band** :
 - Lệnh của FTP : cổng 21
 - Dữ liệu: cổng 20
- NSD phải đăng nhập trước khi truyền file
- Một số server cho phép NSD với tên là anonymous



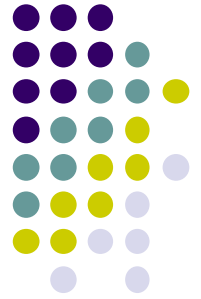
Lệnh và mã trả lời

Một số ví dụ

- **USER *username***
- **PASS *password***
- **LIST** : trả về danh sách file
- **RETR filename** Lấy file
- **STOR filename** Đặt file lên máy chủ

Ví dụ về mã trả lời

- **331 Username OK, password required**
- **125 data connection already open; transfer starting**
- **425 Can't open data connection**
- **452 Error writing file**



Ví dụ về ftp client

Command line

C:\Documents and Settings\hongson>ftp

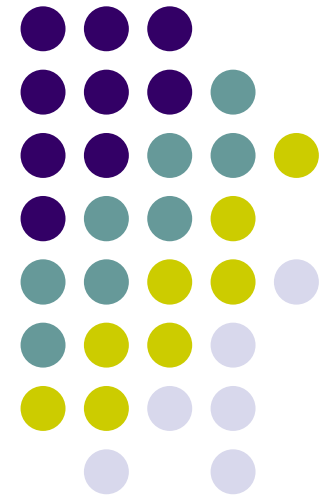
ftp> ?

Commands may be abbreviated. Commands are:

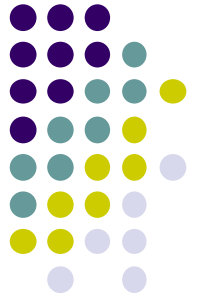
!	delete	literal	prompt	send
?	debug	ls	put	status
append	dir	mdelete	pwd	trace
ascii	disconnect	mdir	quit	type
bell	get	mget	quote	user
binary	glob	mkdir	recv	verbose
bye	hash	mls	remotehelp	
cd	help	mput	rename	
close	lcd	open	rmdir	

GUI FTP clients: IE, Firefox, GFTP,

Domain name service



Giới thiệu chung



- Tên miền: định danh trên tầng ứng dụng cho các nút mạng
 - Trên Internet được quản lý tập trung
 - Quốc tế: ICANN
 - Việt Nam: VNNIC
- DNS(Domain Name System): hệ thống tên miền gồm các máy chủ quản lý thông tin tên miền và cung cấp dịch vụ DNS
- Vấn đề phân giải tên miền sang địa chỉ IP
 - Người sử dụng dùng tên miền để truy cập dịch vụ
 - Máy tính và các thiết bị mạng không sử dụng tên miền mà dùng địa chỉ IP khi trao đổi dữ liệu
- Làm thế nào để chuyển đổi tên miền sang địa chỉ IP?

Chuyển đổi địa chỉ và ví dụ



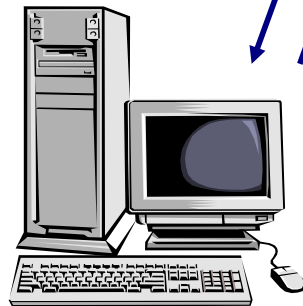
- Máy tính dùng địa chỉ IP
- NSD dùng tên miền



NSD

Tôi muốn vào địa chỉ
www.soict.hust.edu.vn

↓
Cần có chuyển
đổi địa chỉ



Máy chủ web
202.191.56.65

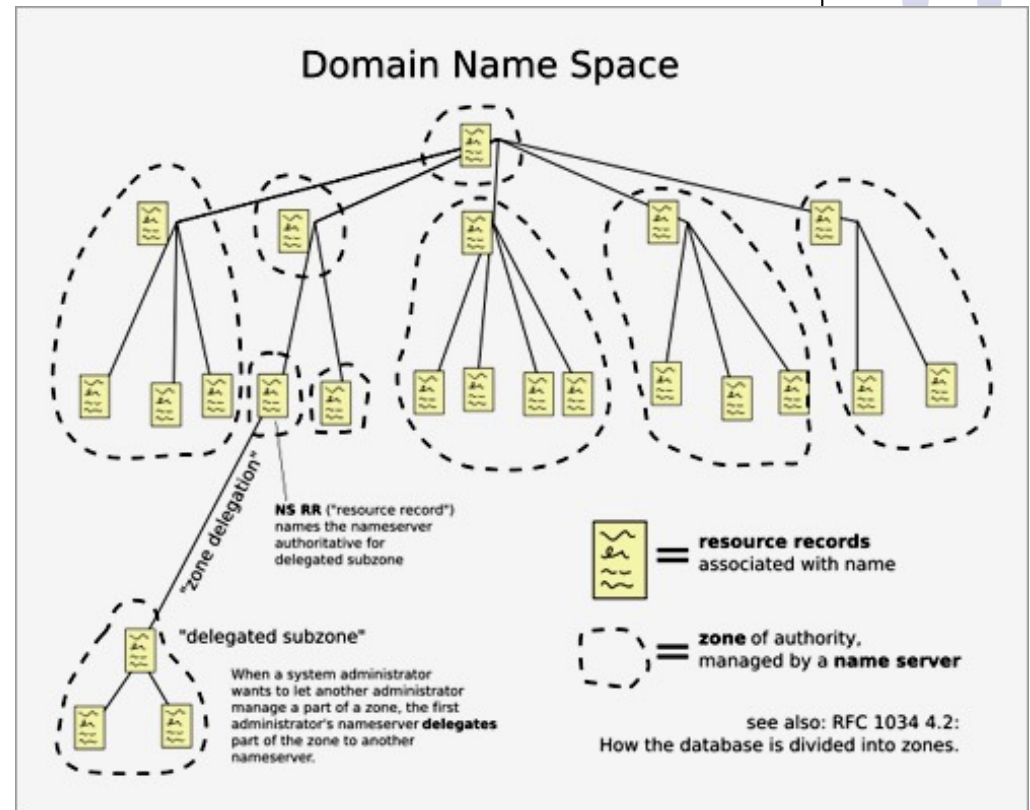
Mời truy cập vào
202.191.56.65



Máy chủ tên
miền

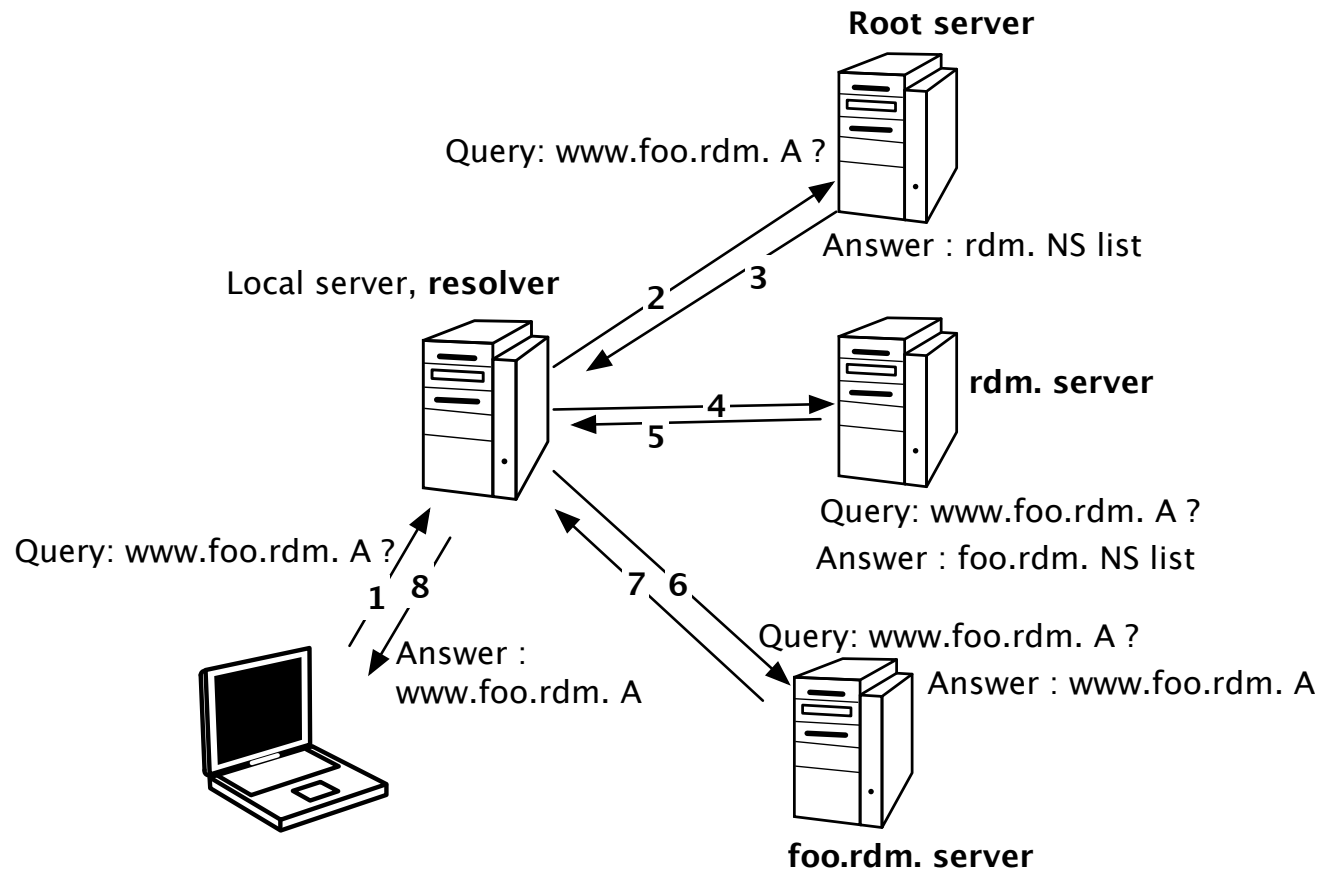
Không gian tên miền

- Kiến trúc : hình cây
 - Root: Nút gốc
 - Chia thành các zone
- Mỗi nút là một tập hợp các bản ghi mô tả tên miền tương ứng với nút đó. Ví dụ:
 - SOA: Bản ghi cung cấp thông tin về một zone.
 - NS: Ủy quyền 1 zone cho một máy chủ
 - A: Bản ghi chứa ánh xạ địa chỉ

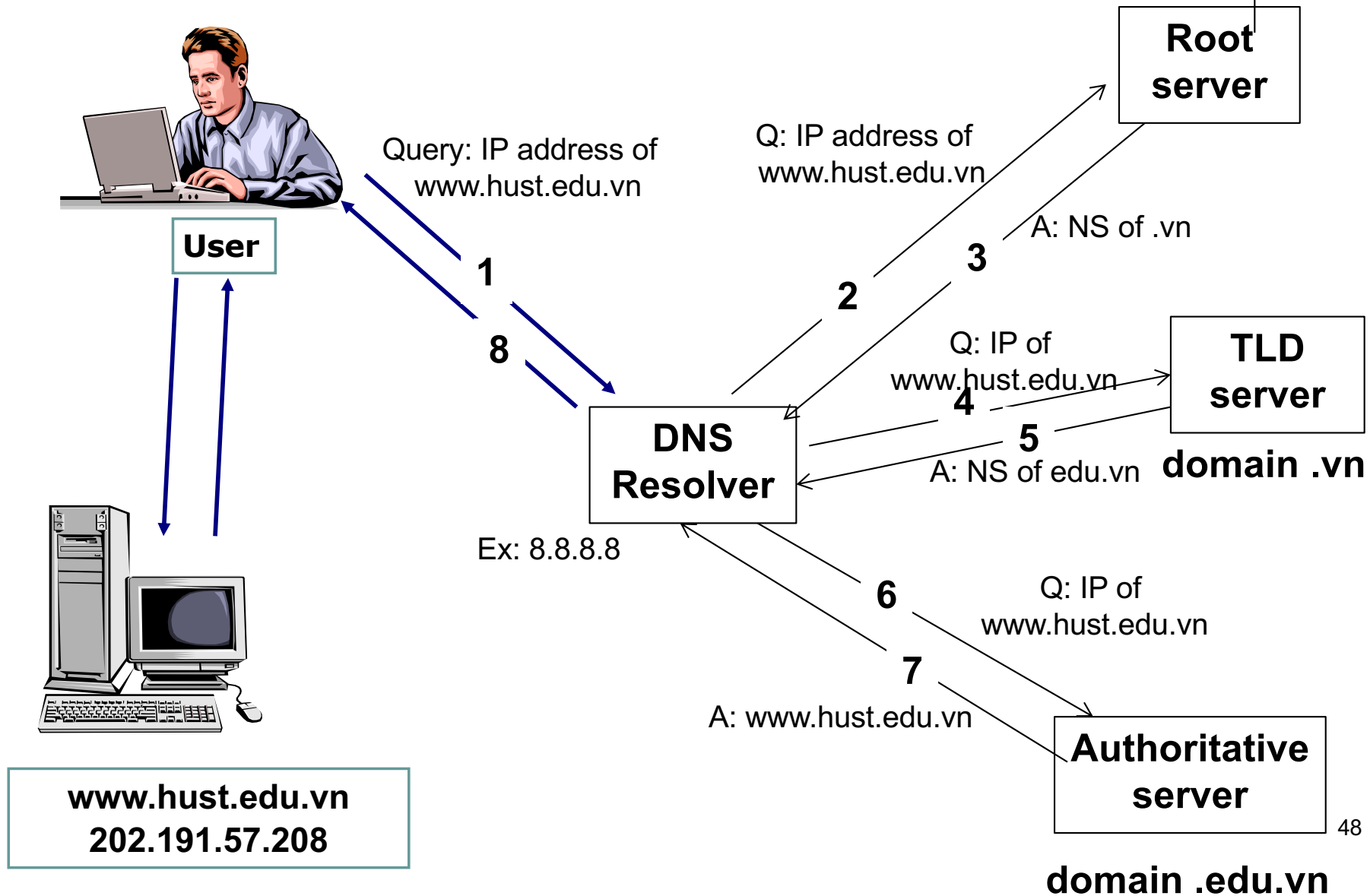


Hình ảnh từ:
Wikipedia

Domain Name Service



Domain Name Service

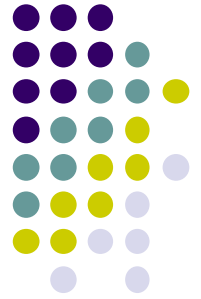




Domain name service

- **Root server:**

- Manage the list of TLD nameserver based on the extension of that domain (.com, .net, .org, etc.)
- The root nameservers are overseen by a nonprofit called the Internet Corporation for Assigned Names and Numbers (ICANN).
- There are 13 root server but there are multiple copies of each one all over the world



Domain name service

- **Top Level Domain (TLD) nameserver**
 - maintains information for all the domain names that share a common domain extension
 - Generic top-level domains: .com, .org, .net, .edu, and .gov.
 - Country code top-level domains: .vn, .uk, .fr, .jp ...
 - If a user searches for google.com, after receiving a response from a root nameserver, the recursive resolver then sends a query to a .com TLD nameserver.
- Management of TLD nameservers is handled by IANA, a branch of ICANN



Dmain name service

- **Authoritative domain name server**
 - Final holder of the DNS record
 - Possible records:
 - A: address, AAAA: IPv6 address
 - NS: name server
 - MX: mail exchange
 - CNAME: alias
 - PTR: address to name
 - SRV: generic service (used for SIP)
 - SOA: start of authority (gives various info. about zone)
 - TXT: text




Phân giải tên miền

- Tụ phân giải
 - File HOST:
 - Windows: C:\WINDOWS\system32\drivers\etc\
 - Linux: /etc/hosts
 - Bộ đệm của ứng dụng
- Dịch vụ phân giải tên miền: client/server
 - Giao thức tầng ứng dụng: DNS
 - Sử dụng dịch vụ UDP/TCP với cổng dịch vụ là 53
 - Phân giải đệ quy (Recursive Query)
 - Phân giải tương tác (Interactive Query)

Thông điệp DNS


- DNS Query và DNS Reply: Chung khuôn dạng
- Identification: Định danh của truy vấn
 - Thông điệp trả lời phải có giá trị Identification trùng với thông điệp truy vấn
- Flags: Các cờ điều khiển
- #Question: Số lượng tên miền được truy vấn
- QUESTION: các tên miền được truy vấn



Identification	Flags
#Question	#Answer RRs
#Authority RRs	#Additional RRs
QUESTION	
ANSWER	
AUTHORITY	
ADDITIONAL	

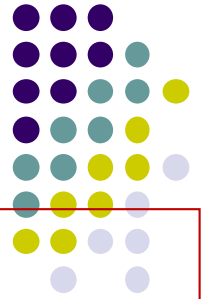
Thông điệp DNS

- #Answer RRs: Số lượng bản ghi trả lời
- ANSWER: Các bản ghi trả lời
- # Authority RRs: Số lượng bản ghi từ các máy chủ được ủy quyền khác
- AUTHORITY: Các bản ghi của máy chủ được ủy quyền khác
- #Additional RRs: Số lượng các bản ghi bổ sung
- ADDITIONAL: Các bản ghi bổ sung



Identification	Flags
#Question	#Answer RRs
#Authority RRs	#Additional RRs
QUESTION	
ANSWER	
AUTHORITY	
ADDITIONAL	

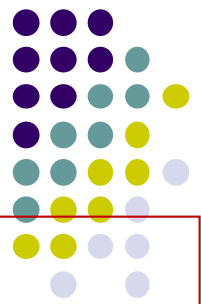
Ví dụ: dig linux.com



```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 3
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

TTL: thời gian(s) lưu giữ
trả lời trong cache

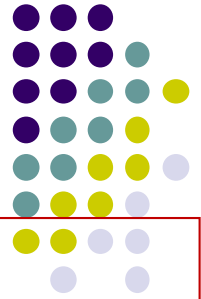
Ví dụ: dig linux.com –thực hiện truy vấn DNS



```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 3
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

Tên các máy chủ DNS server trả lời truy vấn.
Nếu phần ANSWER rỗng, DNS Resolver gửi truy vấn tới các máy chủ này

Ví dụ: dig linux.com

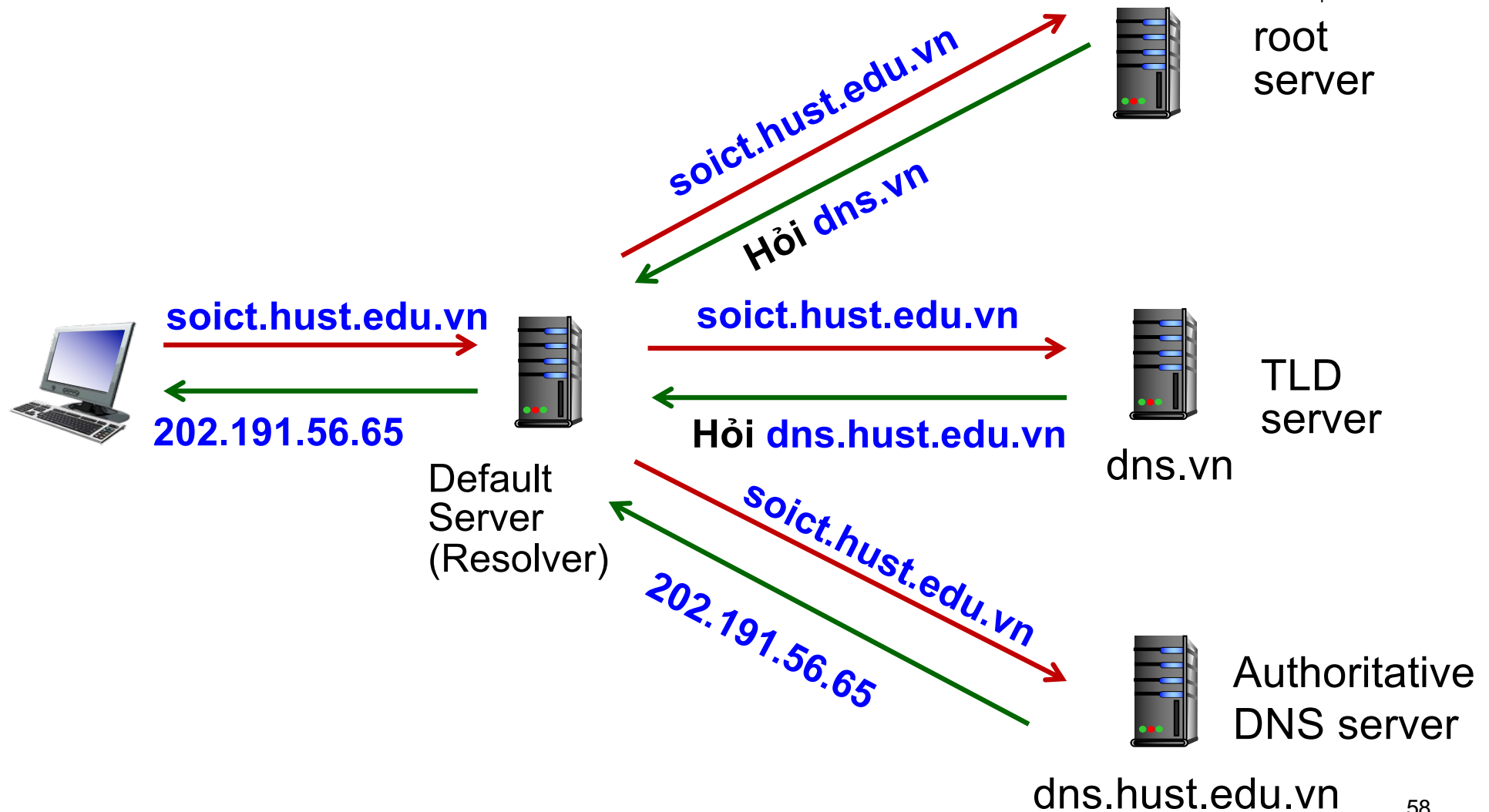


```
; <> DiG 9.9.2-P1 <> linux.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21655
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2,
ADDITIONAL: 3
;; QUESTION SECTION:
;linux.com. IN A
;; ANSWER SECTION:
linux.com. 1786 IN A 140.211.167.51
linux.com. 1786 IN A 140.211.167.50
;; AUTHORITY SECTION:
linux.com. 86386 IN NS ns1.linux-foundation.org.
linux.com. 86386 IN NS ns2.linux-foundation.org.
;; ADDITIONAL SECTION:
ns1.linux-foundation.org. 261 IN A 140.211.169.10
ns2.linux-foundation.org. 262 IN A 140.211.169.11
```

Địa chỉ IP của các máy chủ trả lời truy vấn.
Thông tin này được lưu vào cache

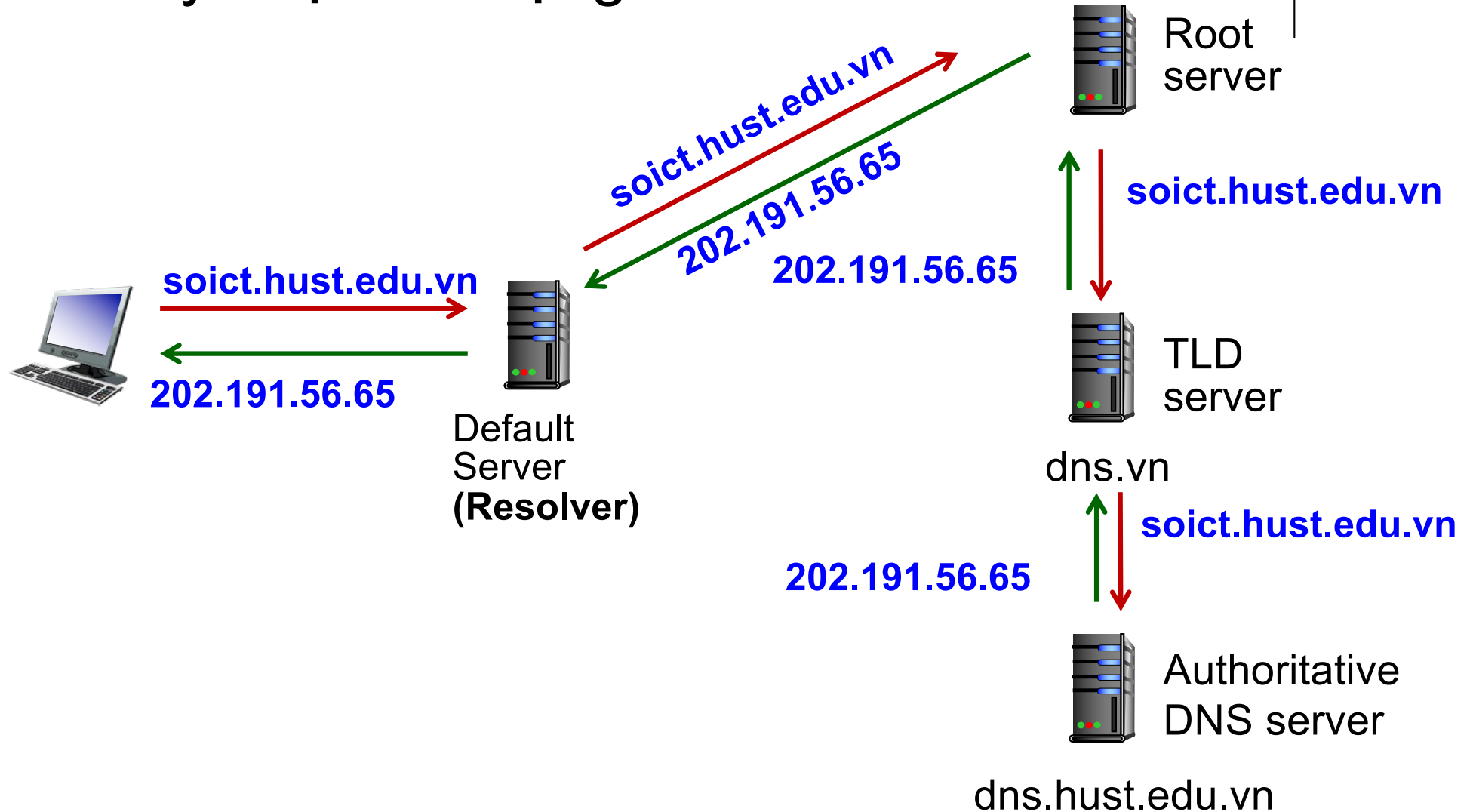
Phân giải tương tác

- Cơ chế mặc định trên các máy chủ DNS



Phân giải đệ quy

- Tùy chọn mở rộng





Tóm tắt

- Mô hình ứng dụng
 - Client-server vs. P2P
- Một số ứng dụng và giao thức
 - HTTP
 - Mail
 - FTP
- Về nhà, hãy tìm hiểu thêm
 - P2P
 -
- Giao diện lập trình Socket



HTTPS

- HTTPS
- Cơ bản về ATTT
 - Assymestric system: Public key – Private key
 - Chứng chỉ số (certificate)
 - Symmestric system
 - Chữ ký điện tử
- Ví dụ ứng dụng trong HTTPS.



Acknowledgment

- Bài giảng có sử dụng các tư liệu và hình vẽ từ:
 - Tài liệu của trường đại học Keio và Ritsumeikan
 - Tài liệu “Computer Network, a top down approach” của J.F Kurose và K.W. Ross