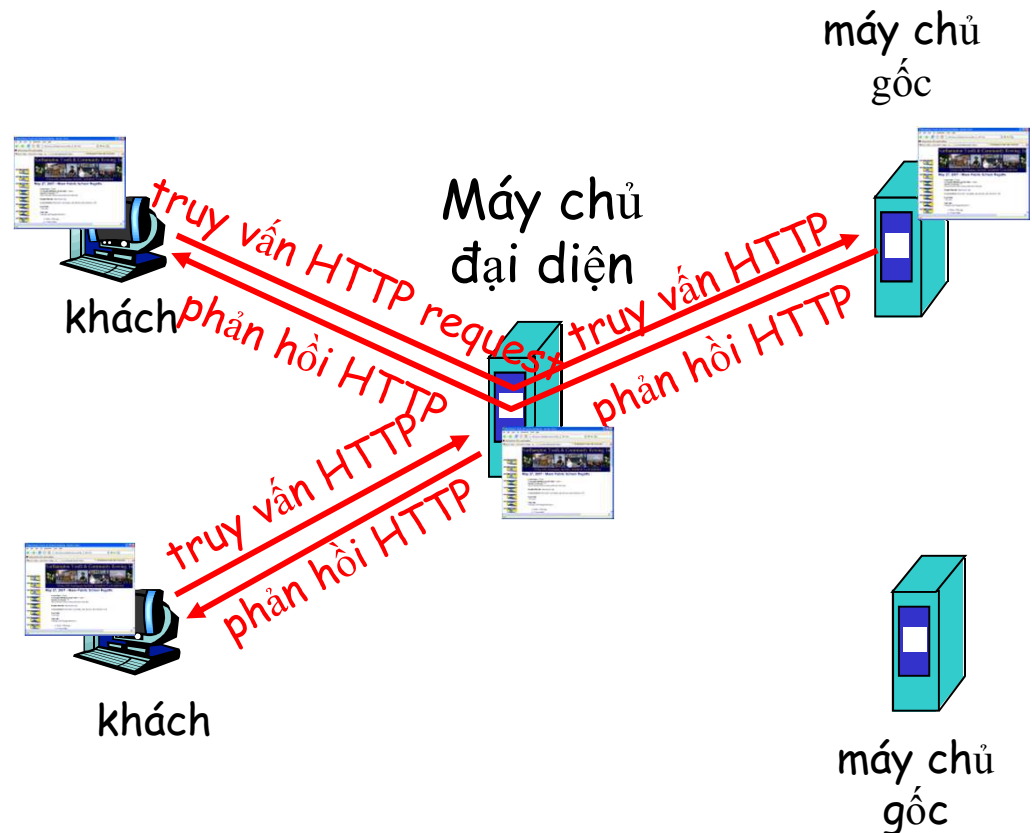


Bộ nhớ đệm (BNĐ) Web (máy chủ đại diện - proxy server)

Mục đích: đáp ứng yêu cầu của n/d mà không cần máy chủ gốc

- thiết lập trình duyệt: truy cập web qua bộ nhớ đệm
- trình duyệt gửi toàn bộ truy vấn HTTP tới máy chủ đại diện
 - ❖ đối tượng có trong bnđ: bnđ gửi trả đối tượng
 - ❖ ngoài ra, bnđ sẽ truy vấn đối tượng từ máy chủ gốc, sau đó gửi lại cho người dùng, đồng thời lưu lại trong bnđ



Bộ nhớ đệm Web (tt)

- ❑ bnd làm việc vừa như khách vừa như chủ
- ❑ thường thì bnd thường được cài đặt bởi nccdvd (trường đại học, cơ quan)

Tại sao cần có bnd Web?

- ❑ giảm thời gian phản hồi cho truy vấn n/dùng
- ❑ giảm lưu lượng trong một đường kết nối của công sở.
- ❑ cho phép các nccdvd tăng thêm nội dung cung cấp cho người dùng

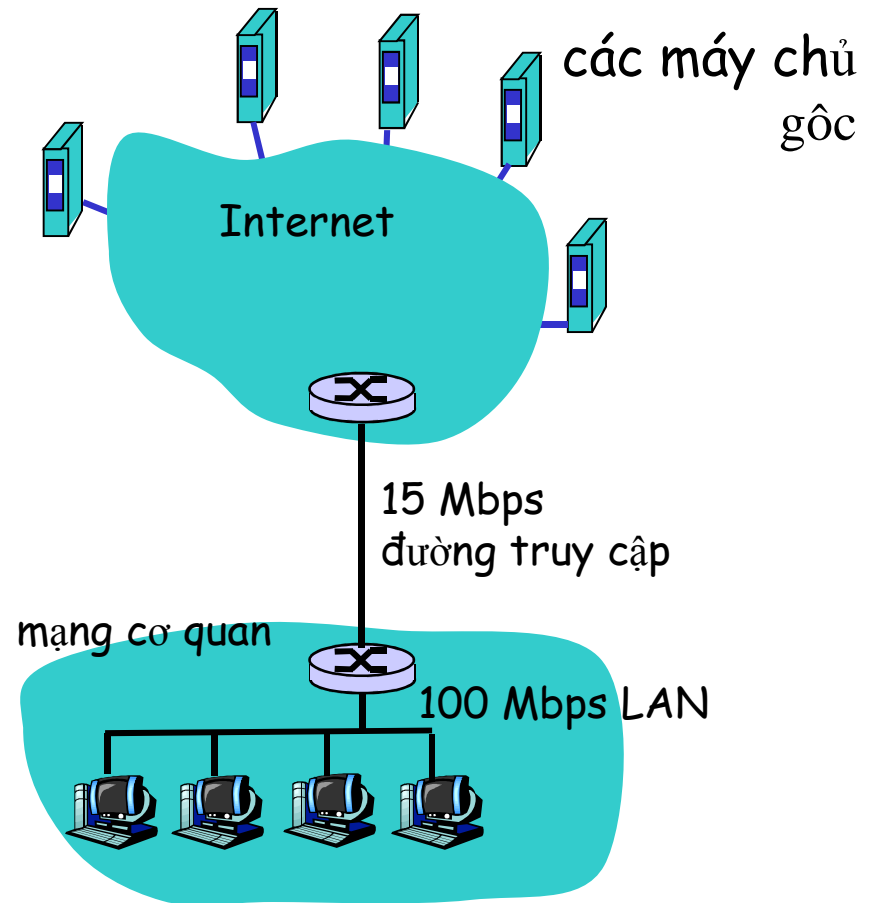
Ví dụ hiện thực bnd

Giả thiết

- ❑ kích thước trung bình của đối tượng = 1,000,000 bit
- ❑ tần số t/bình của truy vấn từ trình duyệt của cơ quan tới máy chủ gốc = 15 lần/s
- ❑ độ trễ xoay vòng từ bất cơ quan tới bất kì máy chủ gốc nào = 2 s

Kết quả

- ❑ h/suất sử dụng LAN = 15%
- ❑ h/suất sử dụng trên đường truy cập = 100%
- ❑ độ trễ tổng = độ trễ Internet + độ trễ truy cập + độ trễ LAN
= 2 s + vài phút + vài ms



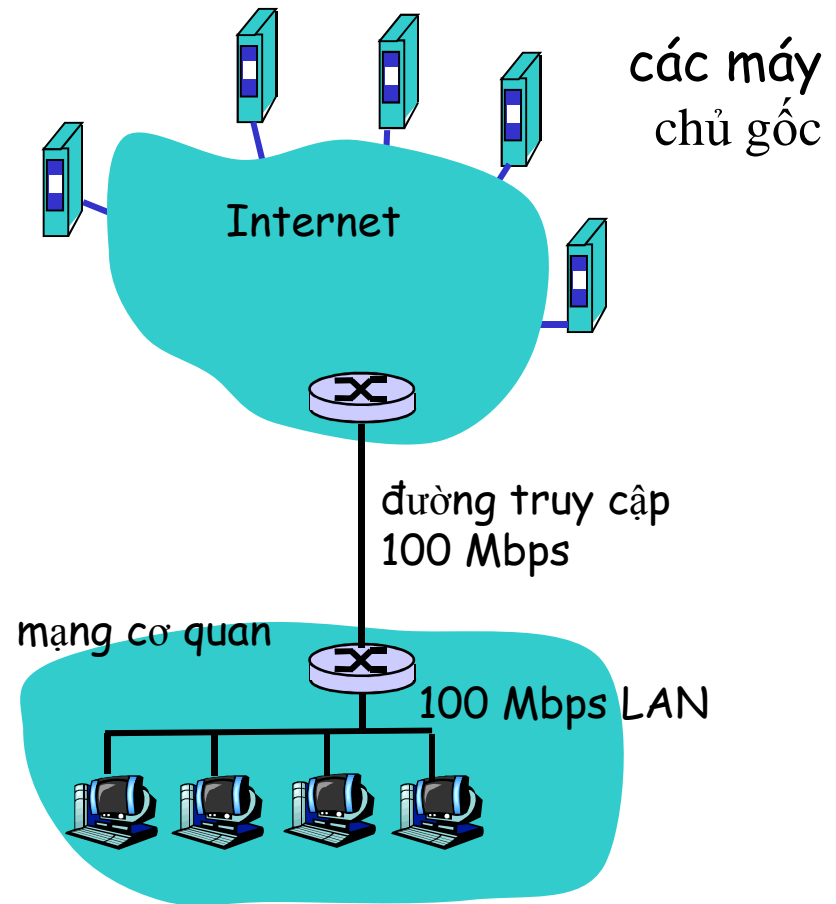
Ví dụ hiện thực bnd (tt)

những giải pháp khả dĩ

- tăng băng thông của đường truy cập lên khoảng 100 Mbps

consequence

- h/suất sử dụng LAN = 15%
- h/suất sử dụng đường truy cập = 15%
- độ trễ tổng = độ trễ Internet + độ trễ truy cập + độ trễ LAN
- = 2 s + ms + ms
- rất tốn kém khi nâng cấp



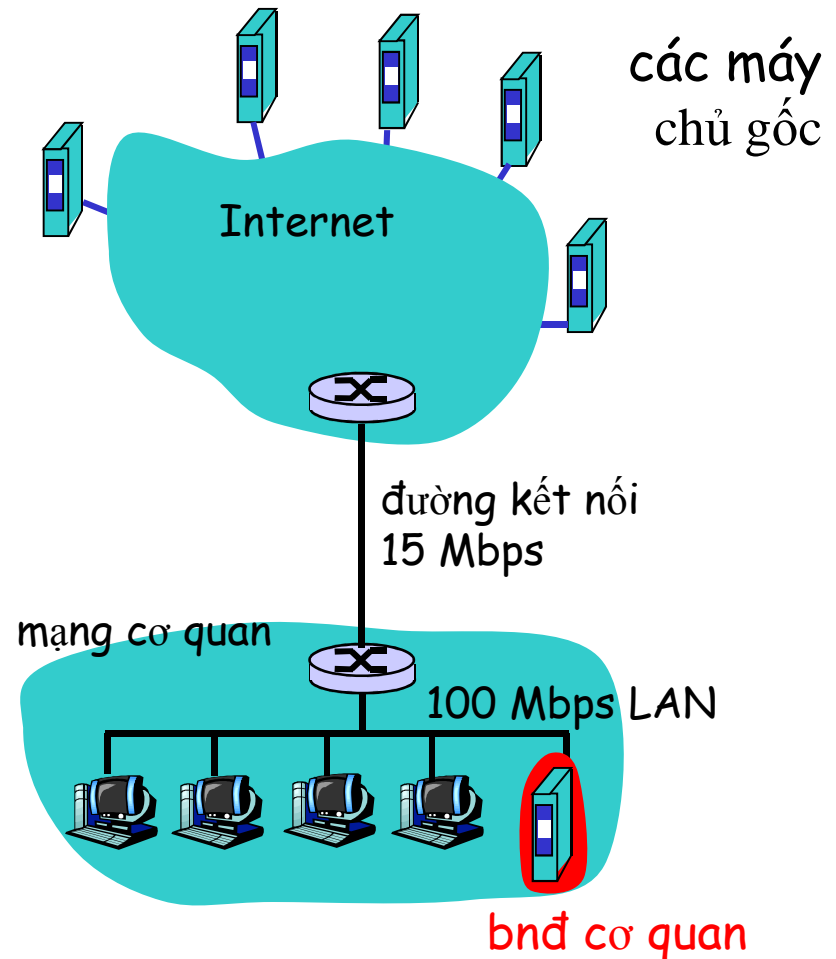
Ví dụ hiện thực bnd (tt)

giải pháp: cài đặt bnd

- giả sử tần số truy cập bnd là 0.4

kết quả:

- 40% truy vấn sẽ được đáp ứng ngay lập tức bởi bnd
- 60% truy vấn sẽ được đáp ứng bởi máy chủ gốc
- hiệu suất sử dụng đường kết nối giảm xuống 60%, làm giảm thời gian trễ xuống thấp (10 msec)
- tổng độ trễ tối đa = độ trễ Internet + độ trễ truy cập + độ trễ LAN = $.6 * (2.01) \text{ secs} + \text{msec} < 1.4 \text{ secs}$



GET có điều kiện

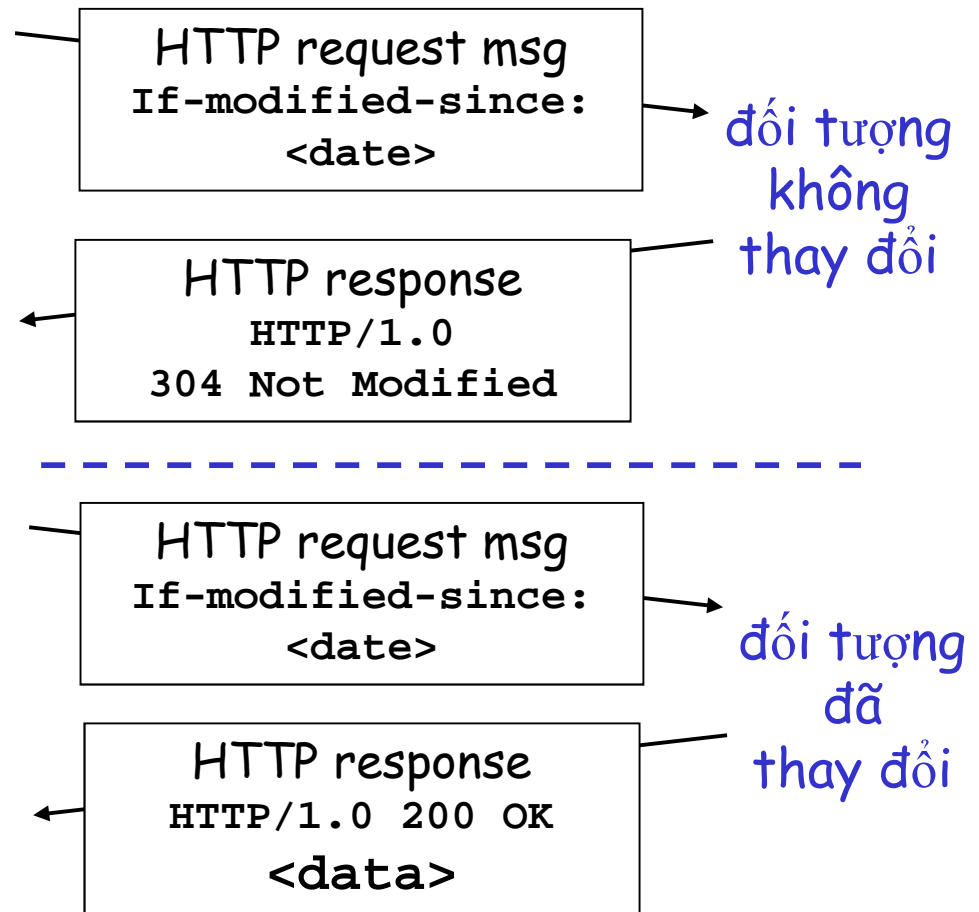
Mục đích: không gửi đối tượng nếu như bnd đã có bản cập nhật mới nhất của nó

□ bnd: chỉ rõ thời gian của bản lưu bnd trong truy vấn HTTP
If-modified-since:
<date>

□ máy chủ: sẽ không gửi lại đối tượng được yêu cầu nếu bản lưu là bản cập nhật mới nhất:
HTTP/1.0 304 Not
Modified

bnd

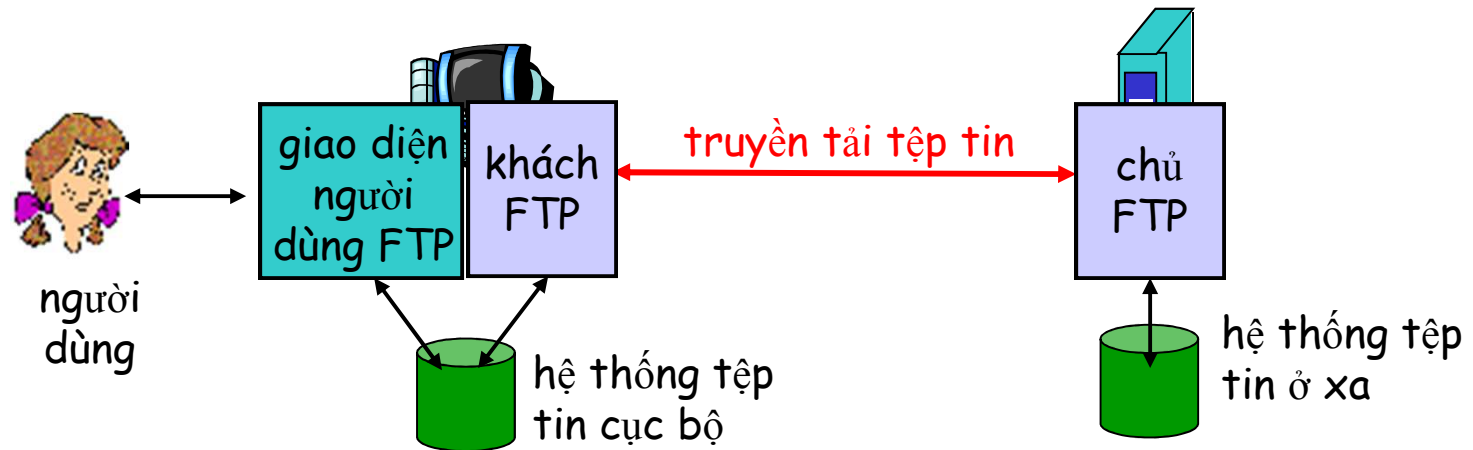
máy chủ



Chapter 2: Application layer

- ❑ 2.1 Principles of network applications
- ❑ 2.2 Web and HTTP
- ❑ 2.3 FTP
- ❑ 2.4 Electronic Mail
 - ❖ SMTP, POP3, IMAP
- ❑ 2.5 DNS
- ❑ 2.6 P2P applications
- ❑ 2.7 Socket programming with TCP
- ❑ 2.8 Socket programming with UDP
- ❑ 2.9 Building a Web server

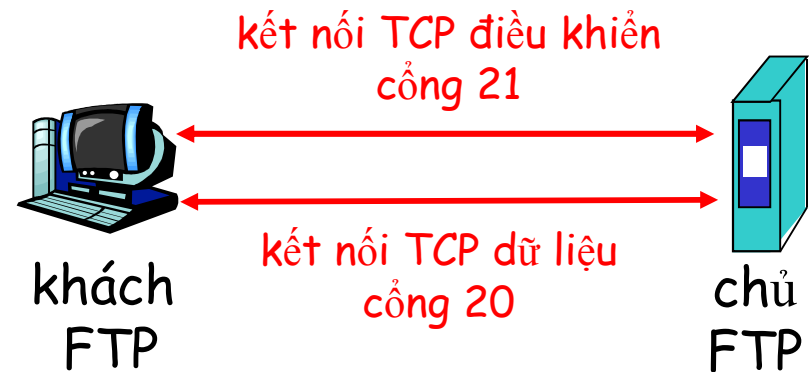
FTP: Giao thức truyền tải tệp tin



- ❑ truyền tải tệp tin đến/từ máy ở xa
- ❑ mô hình khách/chủ
 - ❖ *khách*: là phía bắt đầu quá trình truyền tải
 - ❖ *chủ*: máy ở xa
- ❑ ftp: RFC 959
- ❑ máy chủ ftp: cổng 21

FTP: điều khiển riêng biệt, kết nối dữ liệu

- ❑ Khách FTP kết nối tới máy chủ FTP tại cổng 21, giao thức truyền tải là TCP
- ❑ khách được kiểm tra danh tính thông qua kết nối điều khiển
- ❑ khách duyệt các thư mục trên máy từ xa bằng cách gửi các câu lệnh thông qua kết nối điều khiển.
- ❑ khi chủ nhận được câu lệnh truyền tải tệp, chủ mở kết nối TCP thứ 2 tới khách
- ❑ sau khi truyền tải xong 1 tệp, chủ đóng kết nối dữ liệu.



- ❑ chủ mở một kết nối TCP dữ liệu khác để truyền tải một tệp khác.
- ❑ điều khiển kết nối: "**ngoại tuyến**" (out of band)
- ❑ máy chủ FTP lưu lại "trạng thái": thư mục hiện tại, thông tin về danh tính

Các câu lệnh và phản hồi FTP

Ví dụ câu lệnh:

- ❑ gửi đi dưới dạng văn bản ASCII qua kết nối điều khiển
- ❑ `USER tên_người_dùng`
- ❑ `PASS mật_khẩu`
- ❑ `LIST` liệt kê danh sách các tệp trong thư mục hiện tại
- ❑ `RETR tên_tệp` tải tệp tin về
- ❑ `STOR tên_tệp` tải tệp tin lên máy từ xa

Ví dụ mã trả về

- ❑ mã trạng thái và mô tả (như trong HTTP)
- ❑ 331 Username OK, password required
- ❑ 125 data connection already open; transfer starting
- ❑ 425 Can't open data connection
- ❑ 452 Error writing file

Chapter 2: Application layer

- ❑ 2.1 Principles of network applications
- ❑ 2.2 Web and HTTP
- ❑ 2.3 FTP
- ❑ 2.4 Electronic Mail
 - ❖ SMTP, POP3, IMAP
- ❑ 2.5 DNS
- ❑ 2.6 P2P applications
- ❑ 2.7 Socket programming with TCP
- ❑ 2.8 Socket programming with UDP

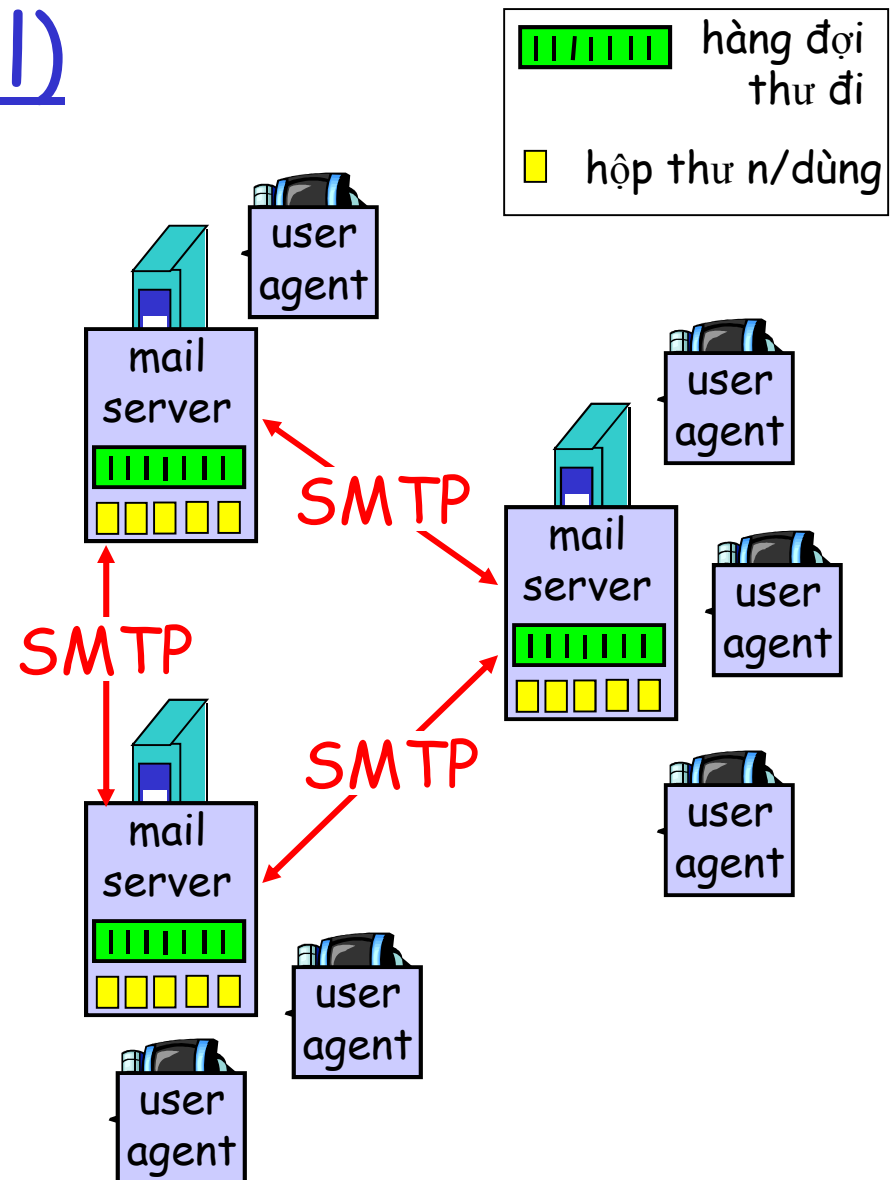
Thư điện tử (e-mail)

Ba thành phần chính:

- ❑ công cụ quản lý thư người dùng (user agent - UA)
- ❑ máy chủ thư
- ❑ giao thức truyền tải thư: SMTP

Công cụ quản lý thư của người dùng

- ❑ soạn thảo, chỉnh sửa, lưu, v.v..
- ❑ vd: Eudora, Outlook, elm, Mozilla Thunderbird
- ❑ thư đi và thư đến được lưu trên máy chủ



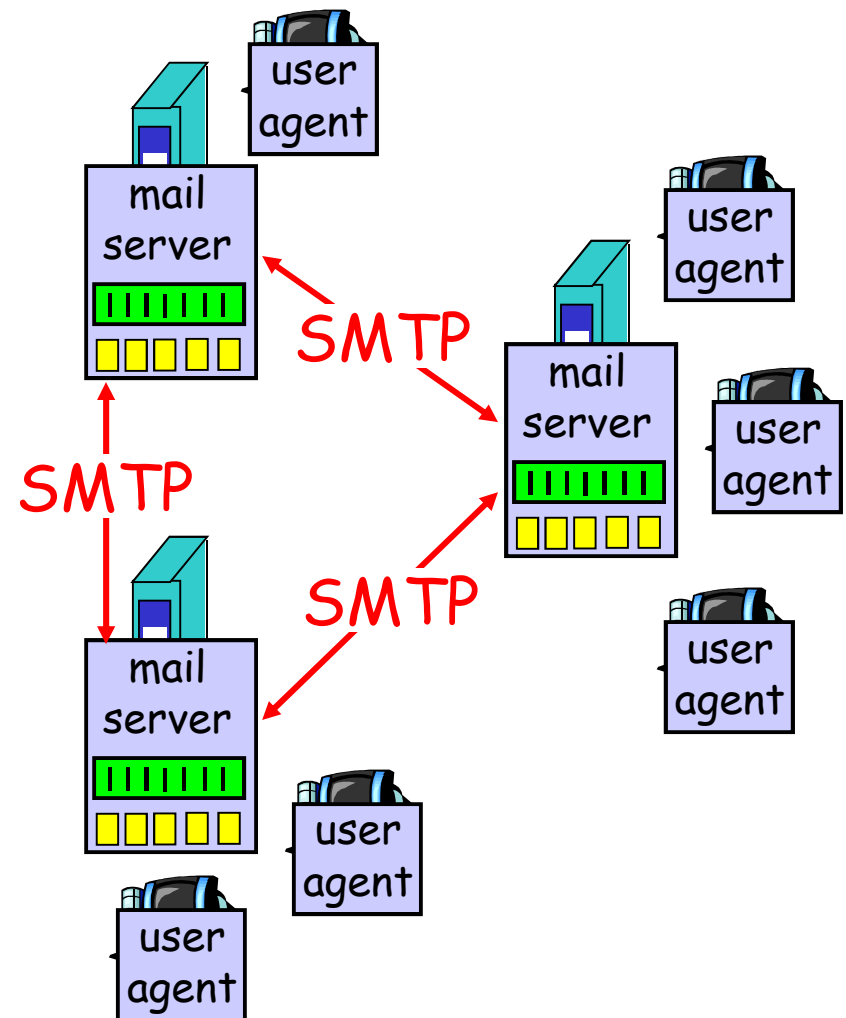
Thư điện tử: máy chủ thư

Máy chủ thư

- **hộp thư** chứa thư đến của người dùng
- **hàng đợi thông điệp** của thư đi

Giao thức SMTP giữa các máy chủ trao đổi thư cho nhau

- ❖ khách: là máy chủ gửi thư
- ❖ "chủ": là máy chủ nhận thư

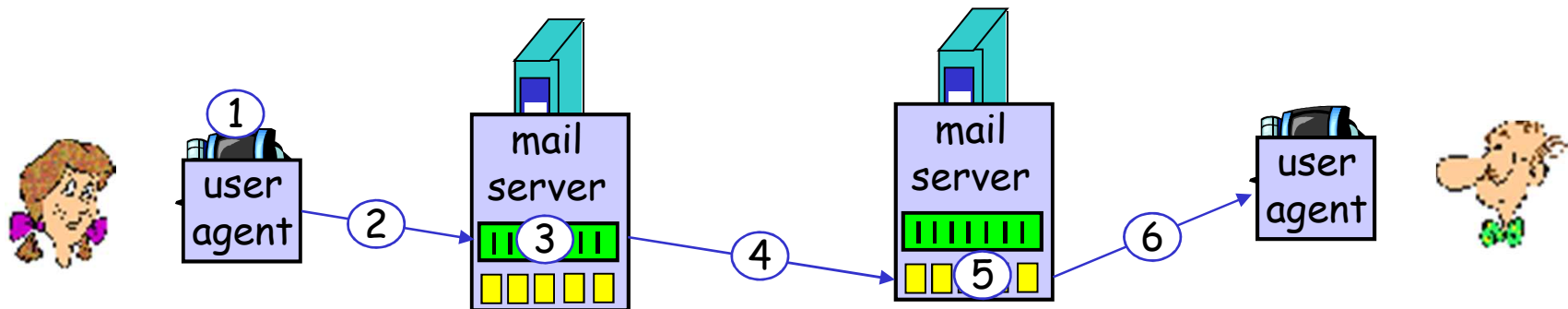


Thư điện tử: SMTP [RFC 2821]

- ❑ sử dụng TCP để gửi thư giữa khách và chủ, cổng 25
- ❑ vận chuyển trực tiếp: từ máy chủ gửi và máy chủ nhận
- ❑ quá trình vận chuyển gồm 3 pha:
 - ❖ bắt tay (chào hỏi)
 - ❖ trao đổi thông điệp
 - ❖ kết thúc
- ❑ tương tác câu lệnh/phản hồi
 - ❖ **câu lệnh:** văn bản ASCII
 - ❖ **phản hồi:** mã trạng thái và mô tả
- ❑ thông điệp phải có định dạng 7-bit ASCII

Kịch bản: Alice gửi thông điệp Bob

- 1) Alice sử dụng UA để soạn thư và gửi tới bob@someschool.edu
- 2) UA của Alice gửi thông điệp tới máy chủ thư của cô ta; thư được đặt trong hàng đợi
- 3) phía khách của SMTP mở kết nối TCP tới máy chủ thư của Bob
- 4) khách SMTP gửi thư của Alice qua kết nối TCP
- 5) máy chủ thư của Bob đặt thư vào hộp thư của Bob
- 6) Bob bật UA của mình lên để đọc thư



Ví dụ tương tác SMTP

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```


Thử nghiệm tương tác SMTP:

- ❑ `telnet tên_máy_chủ 25`
- ❑ nhìn thấy phản hồi 220 từ máy chủ
- ❑ gõ vào các câu lệnh `HELO`, `MAIL FROM`, `RCPT TO`, `DATA`, `QUIT`

bằng cách này bạn có thể gửi thư mà không cần sử dụng công cụ quản lý thư (UA)

SMTP: tổng kết

- ❑ SMTP sử dụng kết nối ổn định
- ❑ SMTP đòi hỏi thông điệp (mào đầu và thân) phải dùng kí tự ASCII 7-bit
- ❑ máy chủ SMTP sử dụng CRLF để xác định điểm kết thúc của thông điệp

So sánh với HTTP:

- ❑ HTTP: tải về
- ❑ SMTP: đẩy đi
- ❑ cả hai đều có tương tác câu lệnh/phản hồi dưới dạng ASCII, các mã trạng thái
- ❑ HTTP: mỗi đối tượng được đóng gói trong thông điệp phản hồi của riêng nó
- ❑ SMTP: nhiều đối tượng được gửi chung trong nhiều phần của một thông điệp

Định dạng thông điệp thư

SMTP: giao thức để trao đổi thư điện tử

RFC 822: chuẩn dùng cho định dạng thông điệp văn bản:

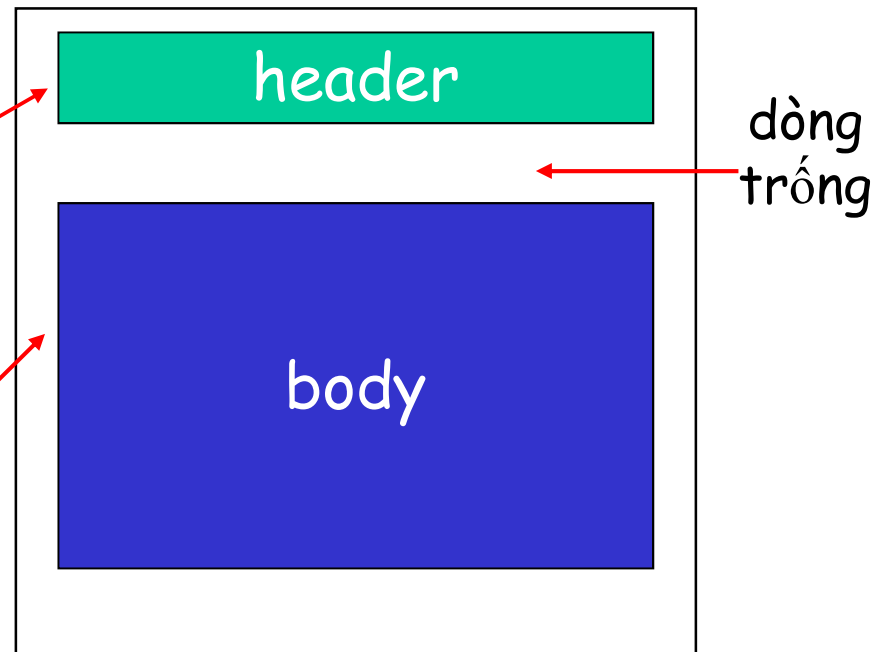
□ các dòng mào đầu, vd:

- ❖ To:
- ❖ From:
- ❖ Subject:

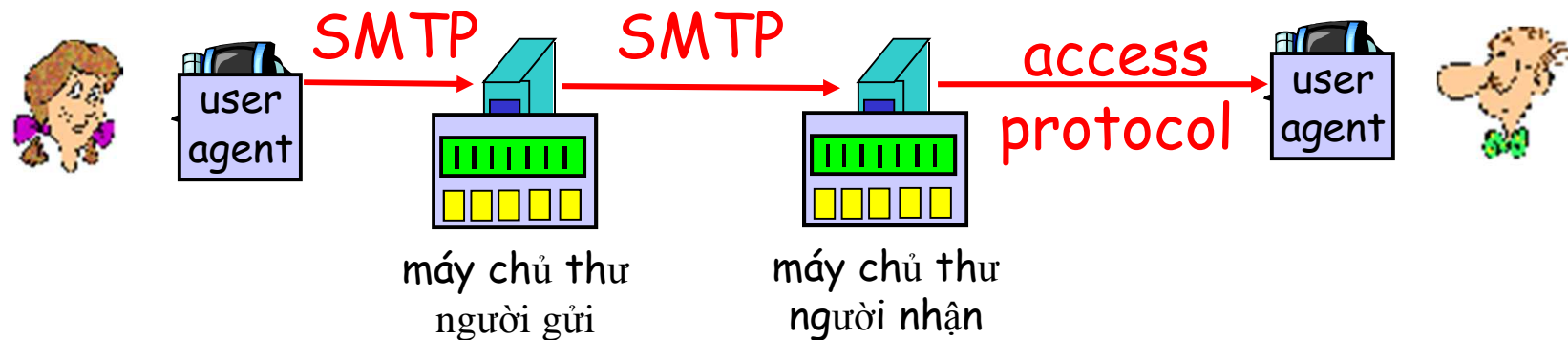
khác nhau phụ thuộc vào câu lệnh SMTP!

□ thân

- ❖ nội dung thư, chỉ chứa kí tự ASCII



Giao thức truy cập thư



- ❑ SMTP: giao/nhận thư đến/từ máy chủ khác
- ❑ Giao thức truy cập thư: lấy thư từ máy chủ
 - ❖ POP: Post Office Protocol [RFC 1939]
 - kiểm tra danh tính (UA<-->máy chủ) và tải về
 - ❖ IMAP: Internet Mail Access Protocol [RFC 1730]
 - nhiều chức năng hơn (phức tạp hơn)
 - quản lý thư lưu trên máy chủ
 - ❖ HTTP: gmail, Hotmail, Yahoo! Mail, v.v..

Giao thức POP3

pha kiểm tra danh tính

- câu lệnh của khách:
 - ❖ user: khai báo tên
 - ❖ pass: mật-khẩu
- phản hồi của chủ
 - ❖ +OK
 - ❖ -ERR

pha giao dịch, khách:

- list: liệt kê số thứ tự thư
- retr: tải thư theo stt
- dele: xóa
- quit: kết thúc

```
S: +OK POP3 server ready
C: user bob
S: +OK
C: pass hungry
S: +OK người dùng đăng nhập thành công

C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK máy chủ POP3 đồng ý ngắt
```

POP3 (tt) và IMAP

POP3

- ❑ ví dụ vừa rồi sử dụng cơ chế "tải-và-xóa".
- ❑ Bob không thể đọc lại thư nếu như anh ta đổi UA
- ❑ "tải-và-giữ lại": bản sao của thư sẽ được lưu trên nhiều UA
- ❑ POP3 là giao thức không trạng thái

IMAP

- ❑ Giữ tất cả thư trong một nơi: trên máy chủ
- ❑ Cho phép n/dùng tổ chức thư theo thư mục
- ❑ IMAP giữ lại trạng thái người dùng qua các phiên làm việc:
 - ❖ tên của thư mục cũng như ánh xạ giữa ID của thông điệp và tên thư mục

Chapter 2: Application layer

- 2.1 Principles of network applications
- 2.2 Web and HTTP
- 2.3 FTP
- 2.4 Electronic Mail
 - ❖ SMTP, POP3, IMAP
- 2.5 DNS
- 2.6 P2P applications
- 2.7 Socket programming with TCP
- 2.8 Socket programming with UDP
- 2.9 Building a Web server

Hệ thống tên miền - DNS (Domain Name System)

Con người: nhiều kí hiệu nhận dạng:

- ❖ tên, số CMND, MSSV

máy tính và thiết bị trên Internet:

- ❖ địa chỉ IP (32 bit) - được sử dụng để liên lạc giữa máy tính
- ❖ "tên máy (hostname)", vd: ww.yahoo.com - được sử dụng bởi con người

Hỏi: ánh xạ giữa địa chỉ IP và tên?

Hệ thống tên miền:

- *có sở dữ liệu phân tán* được hiện thực trong một hệ thống phân cấp của nhiều *máy chủ tên miền*
- *giao thức tầng ứng dụng* cho phép máy tính, bđt và máy chủ tên miền liên lạc với nhau để *phân giải* tên miền (địa chỉ/tên)
- Công việc được đẩy ra các máy đầu cuối (ngoại vi mạng)

DNS

các dịch vụ DNS

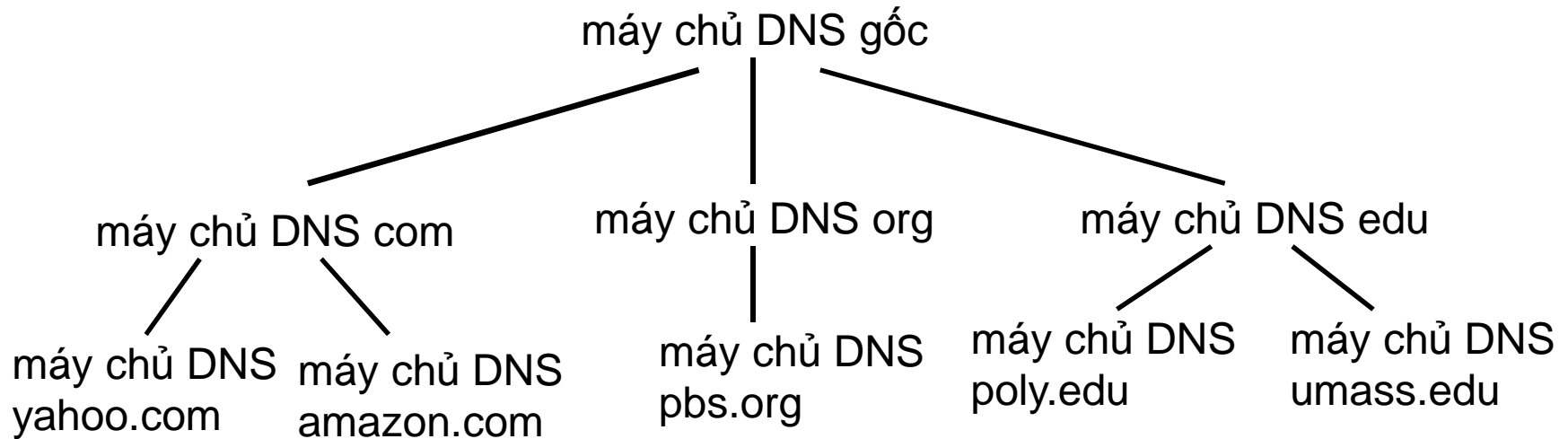
- ❑ phân giải tên miền sang địa chỉ IP
- ❑ đặt tên thay thế cho máy
 - ❖ tên chính thức, và tên thay thế
- ❑ đặt tên thay thế cho máy chủ email
- ❑ phân bổ tải
 - ❖ những trang web lớn thường có nhiều máy chủ với địa chỉ IP khác nhau đằng sau một tên miền chính thức

Tại sao không tập trung DNS?

- ❑ DNS hỏng -> cả mạng ngừng hoạt động
- ❑ quá tải lưu lượng
- ❑ độ trễ cao do đường truyền dài
- ❑ khó bảo trì

không mở rộng *được!*

Cơ sở dữ liệu phân tán và phân cấp

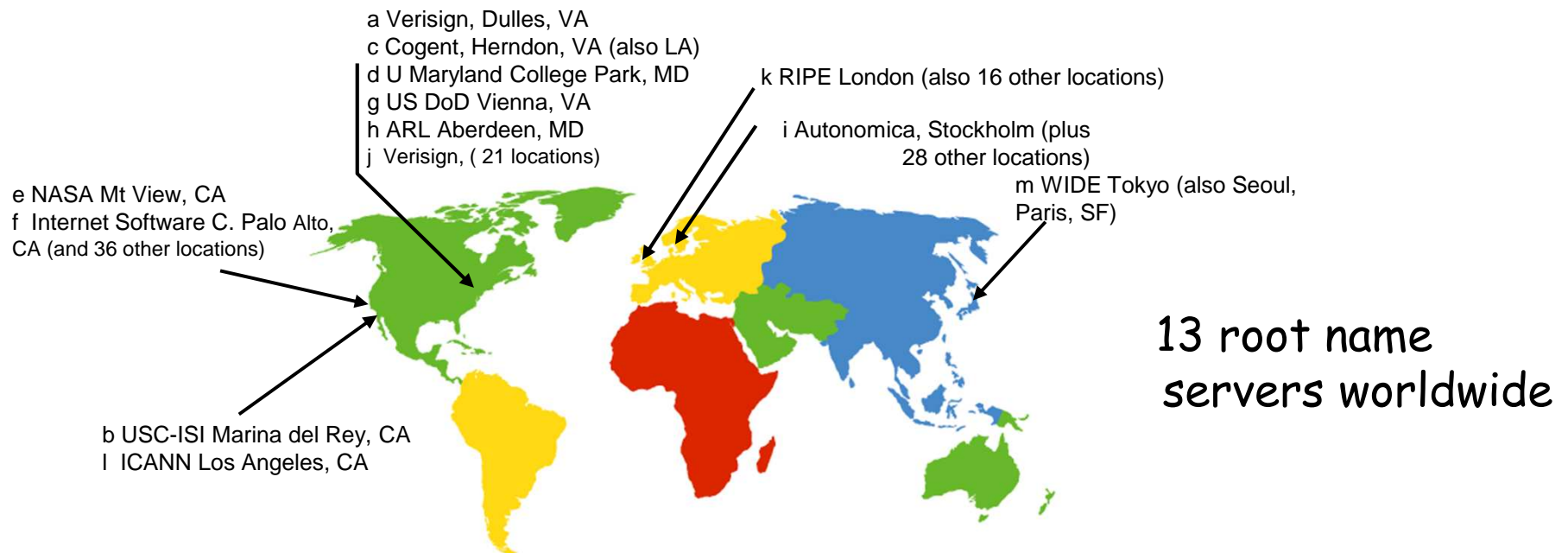


Khách muốn có IP của www.amazon.com:

- ❑ khách truy vấn một máy chủ DNS gốc để tìm ra máy chủ DNS ".com"
- ❑ khách truy vấn máy chủ DNS ".com" để lấy địa chỉ của máy chủ DNS amazon.com
- ❑ khách truy vấn máy chủ DNS amazon.com để lấy địa chỉ IP của www.amazon.com

DNS: Máy chủ gốc

- ❑ các máy chủ tên miền cục bộ sẽ liên hệ máy chủ gốc nếu chúng không thể tự phân giải tên miền
- ❑ máy chủ tên miền gốc:
 - ❖ liên hệ những máy chủ có thẩm quyền nếu không tìm ra ánh xạ
 - ❖ lấy thông tin ánh xạ
 - ❖ trả thông tin ánh xạ về cho các máy chủ tên miền cục bộ



TLD và Máy chủ có thẩm quyền

❑ Máy chủ tên miền cấp cao (Top-level domain - TLD):

- ❖ chịu trách nhiệm cho com, org, net, edu, etc, và tất cả các tên miền quốc gia cấp cao vn, au, ca, jp.
- ❖ Network Solutions duy trì máy chủ com TLD
- ❖ Educause duy trì máy chủ edu TLD

❑ Máy chủ DNS có thẩm quyền:

- ❖ máy chủ DNS cơ quan, cung cấp những ánh xạ tin cậy giữa tên miền và IP cho những máy chủ của chính cơ quan đó (vd: Web, mail).
- ❖ có thể được vận hành bởi cơ quan hay nhà ccdv

Máy chủ tên miền cục bộ

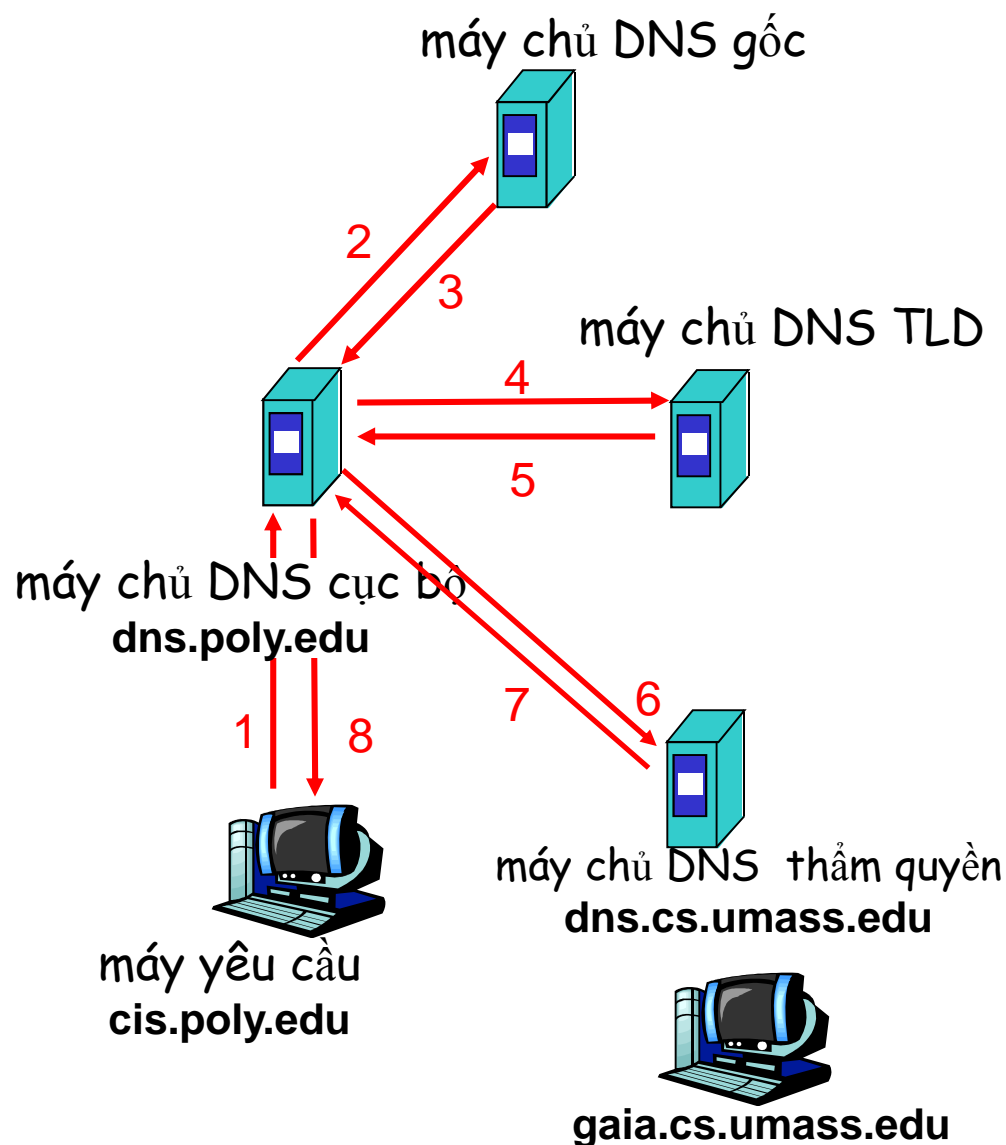
- ❑ không phụ thuộc một cách rõ ràng vào hệ thống phân cấp
- ❑ mỗi ISP (ISP dân sự, cơ quan, trường học) có một máy chủ tên miền cục bộ.
 - ❖ còn được gọi là "máy chủ tên miền mặc định"
- ❑ khi một máy thực hiện một truy vấn DNS, truy vấn sẽ được gửi cho máy chủ DNS cục bộ của nó
 - ❖ hoạt động như là một máy đại diện (proxy), chuyển tiếp truy vấn lên hệ phân cấp

Ví dụ phân giải tên miền DNS

- Máy tại cis.poly.edu muốn có IP của gaia.cs.umass.edu

truy vấn lặp:

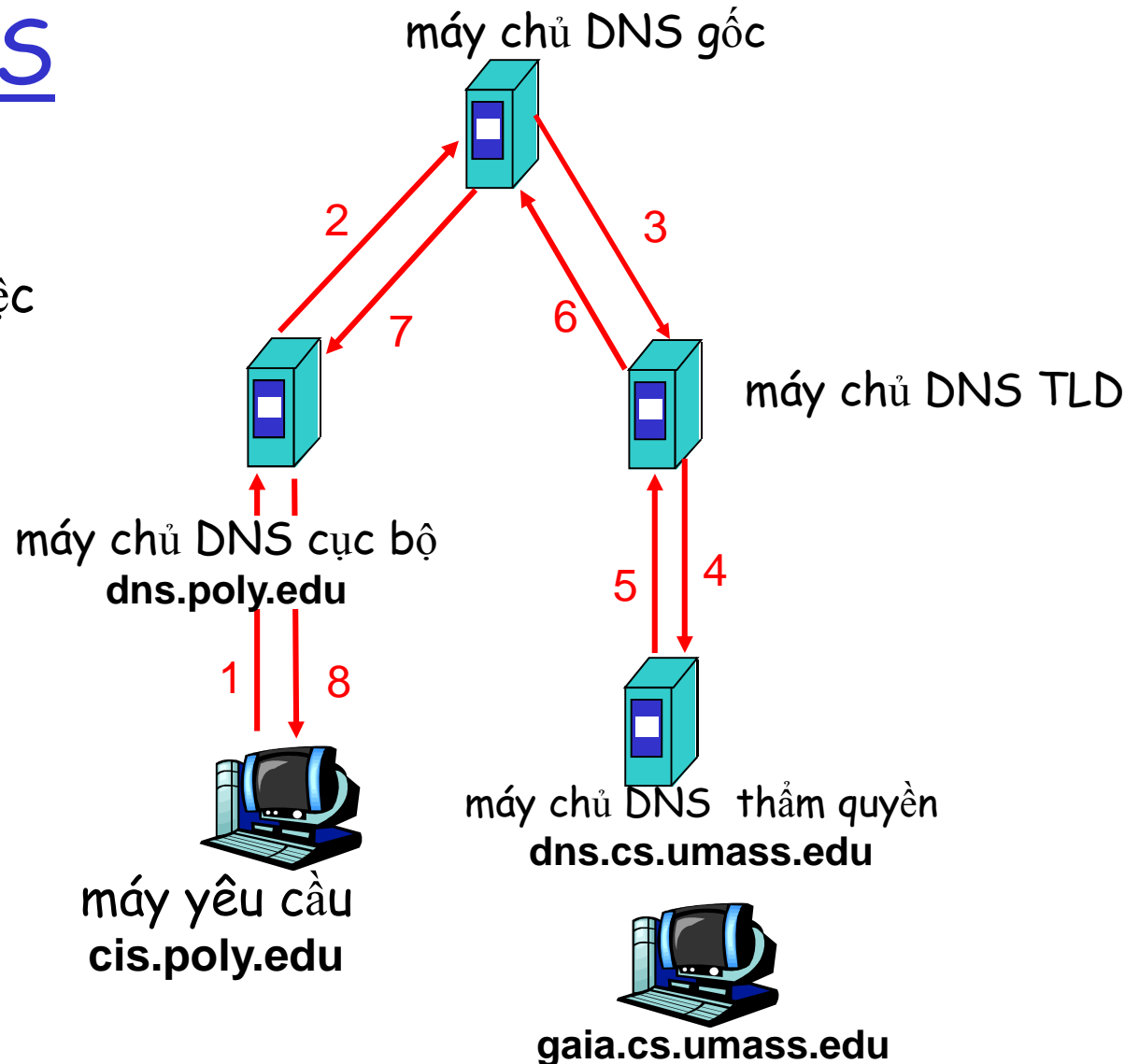
- máy chủ được liên hệ trả về tên của máy chủ khác để liên hệ tiếp
- “Tôi không biết tên miền này, nhưng hãy hỏi máy chủ này xem”



Ví dụ phân giải tên miền DNS

truy vấn đệ qui:

- giao toàn bộ công việc cho máy chủ cấp cao hơn
- tải cao?



DNS: nhớ đệm và cập nhật thẻ ghi

- một khi (bất kì) máy chủ tên miền nhận được giá trị ánh xạ, nó sẽ *nhớ đệm* lại thông tin đó
 - ❖ các thông tin trong bộ nhớ đệm sẽ hết hạn và bị xóa sau một thời gian nhất định
 - ❖ tên các máy chủ TLD thường được nhớ đệm trong máy chủ cục bộ
 - vì vậy máy chủ dns gốc thường ít khi được truy cập
- cơ chế cập nhật/thông báo được thiết kế bởi IETF
 - ❖ RFC 2136
 - ❖ <http://www.ietf.org/html.charters/dnsind-charter.html>