

Chapter 5

Tầng Liên kết dữ liệu (Data Link layer)

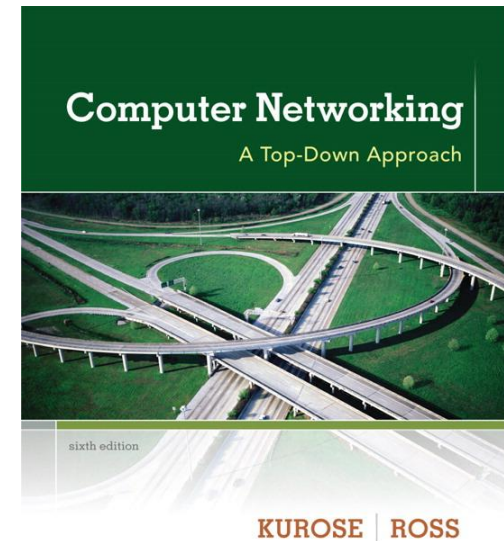
A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a lot of work on our part. In return for use, we only ask the following:

- ❖ If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- ❖ If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

© All material copyright 1996-2012
J.F Kurose and K.W. Ross, All Rights Reserved



*Computer
Networking: A Top
Down Approach
6th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2012*

Chương 5: Tầng Liên kết dữ liệu

Mục tiêu:

- ❖ Hiểu về các nguyên lý của các dịch vụ tầng Liên kết dữ liệu:
 - Phát hiện lỗi và sửa lỗi
 - Chia sẻ kênh broadcast: đa truy cập
 - Định địa chỉ tầng Liên kết dữ liệu
 - Local Area Networks: Ethernet, VLANs
- ❖ Khởi tạo và hiện thực một số công nghệ tầng Liên kết dữ liệu

Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 phát hiện lỗi và sửa lỗi

5.3 các giao thức đa truy cập

5.4 mạng LAN

- Định địa chỉ, ARP
- Ethernet
- switches
- VLANs

5.5 link virtualization: MPLS

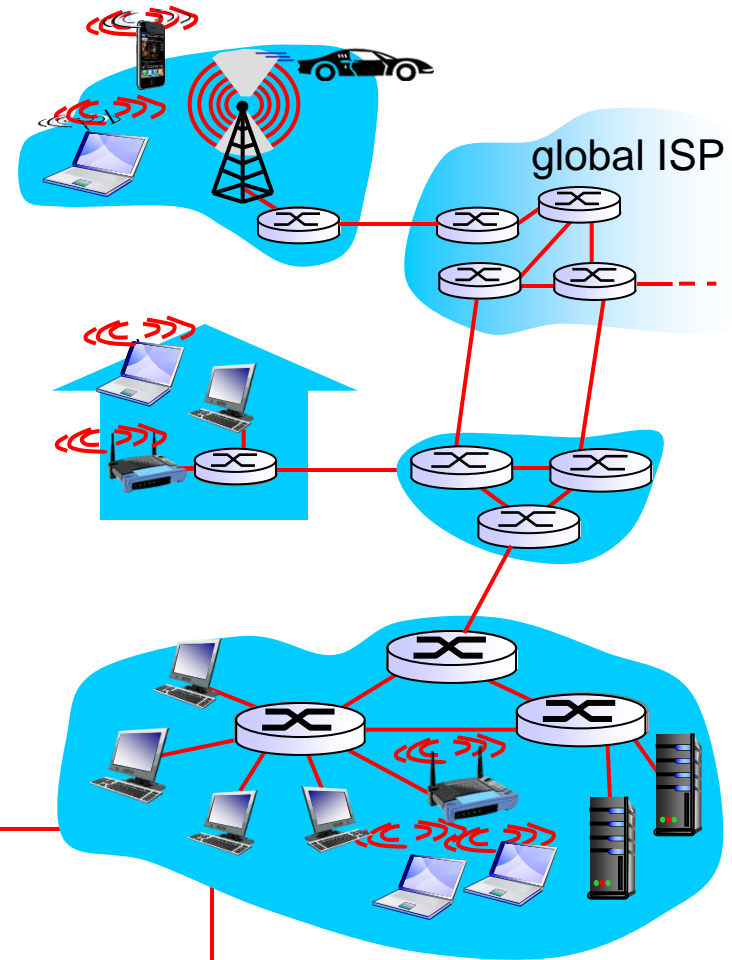
5.6 mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Tầng Liên kết dữ liệu: Giới thiệu

Thuật ngữ:

- ❖ host và router: **node**
- ❖ Các kênh truyền thông kết nối các node lân cận nhau (adjacent nodes) dọc theo đường truyền thông: **links**
 - Kết nối có dây (wired links)
 - Kết nối không dây (wireless links)
 - LANs
- ❖ Gói dữ liệu ở lớp 2: **frame**, đóng gói datagram



Tầng data-link có nhiệm vụ truyền datagram từ 1 node đến node lân cận vật lý (*physically adjacent* node) trên một đường kết nối

Tầng Liên kết dữ liệu: Ngữ cảnh

- ❖ datagram được truyền bởi các giao thức khác nhau thuộc tầng Liên kết dữ liệu trên các đoạn kết nối khác nhau:
 - Ví dụ: Ethernet trên đoạn kết nối thứ 1, frame relay trên các đoạn kết nối trung gian, 802.11 trên đường kết nối cuối cùng
- ❖ Mỗi giao thức tầng Liên kết dữ liệu cung cấp các dịch vụ khác nhau
 - Ví dụ: có thể hoặc không thể cung cấp rdt trên đường kết nối

So sánh:

- ❖ Hành trình từ Princeton đến Lausanne
 - limo: Princeton đến JFK
 - Máy bay: JFK đến Geneva
 - Xe lửa: Geneva đến Lausanne
- ❖ Khách du lịch = **datagram**
- ❖ đoạn đường đi = liên kết truyền thông (**communication link**)
- ❖ Kiểu vận chuyển = **giao thức tầng Liên kết dữ liệu**
- ❖ Đại lý du lịch = **thuật toán định tuyến**

Các dịch vụ tầng Liên kết dữ liệu

- ❖ *Truy cập đường truyền, đóng gói tin (framing):*
 - Đóng gói datagram vào trong frame, thêm header và trailer
 - Điều khiển truy cập kênh truyền nếu môi trường truyền được chia sẻ
 - Các địa chỉ “MAC” được sử dụng trong các header để xác định nguồn và đích
 - Khác với địa chỉ IP!
- ❖ *Truyền tin cậy giữa các node lân cận (adjacent nodes)*
 - Chúng ta đã tìm hiểu làm thế nào để thực hiện điều này ở chương 3!
 - Ít khi được sử dụng trên đường kết nối có tỷ lệ lỗi thấp (cáp quang, một số loại cáp xoắn)
 - Kết nối không dây: tỷ lệ lỗi cao
 - *Hỏi:* lý do cần độ tin cậy ở cả 2 cấp độ liên kết từng chặng và liên kết 2 đầu cuối (end-end)??

Các dịch vụ tầng Liên kết dữ liệu (tt)

❖ *Điều khiển luồng (flow control):*

- Điều khiển tốc độ truyền giữa các node gửi và nhận liên kết nhau

❖ *Phát hiện lỗi (error detection):*

- Lỗi gây ra bởi suy giảm tín hiệu, nhiễu.
- Bên nhận phát hiện sự xuất hiện lỗi:
 - Thông báo bên gửi truyền lại hoặc bỏ frame đó

❖ *Sửa lỗi (error correction):*

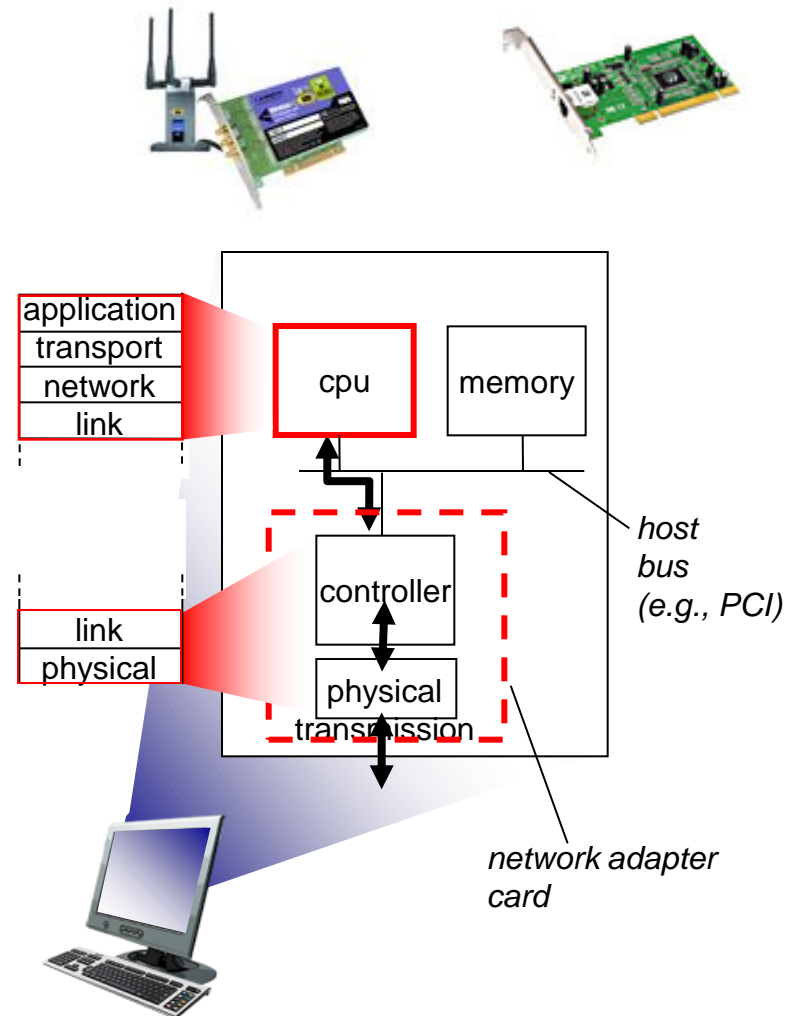
- Bên nhận xác định và **sửa** các bit lỗi mà không cần phải truyền lại

❖ *half-duplex và full-duplex*

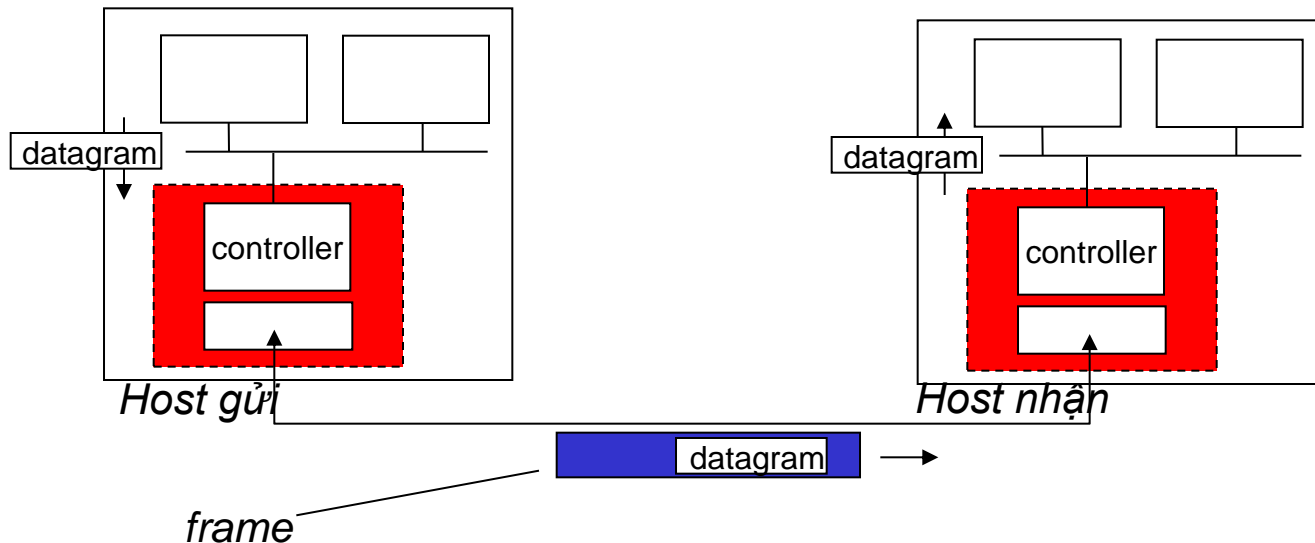
- Các node tại các đầu cuối của mỗi kết nối đều có thể truyền và nhận, nhưng với half duplex thì thao tác này không được thực hiện đồng thời

Tầng Liên kết dữ liệu được triển khai ở đâu?

- ❖ Trong mọi host
- ❖ Tầng Liên kết dữ liệu được triển khai trong “adaptor” (còn gọi là *network interface card* NIC)
 - Ethernet card, 802.11 card; Ethernet chipset
 - Triển khai cả tầng Vật lý và tầng Liên kết dữ liệu
- ❖ Nối vào trong các bus hệ thống của host
- ❖ Sự kết hợp của phần cứng, phần mềm và firmware



Các Adaptor liên lạc



❖ Bên gửi:

- Đóng gói datagram vào trong frame
- Thêm các bit kiểm tra lỗi, rdt và điều khiển luồng...

❖ Bên nhận

- Tìm lỗi, rdt và điều khiển luồng...
- Tách các datagram ra, chuyển lên tầng trên tại nơi nhận

Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 phát hiện lỗi và sửa lỗi

5.3 các giao thức đa truy cập

5.4 mạng LAN

- Định địa chỉ, ARP
- Ethernet
- switches
- VLANs

5.5 link virtualization: MPLS

5.6 mạng trung tâm dữ liệu

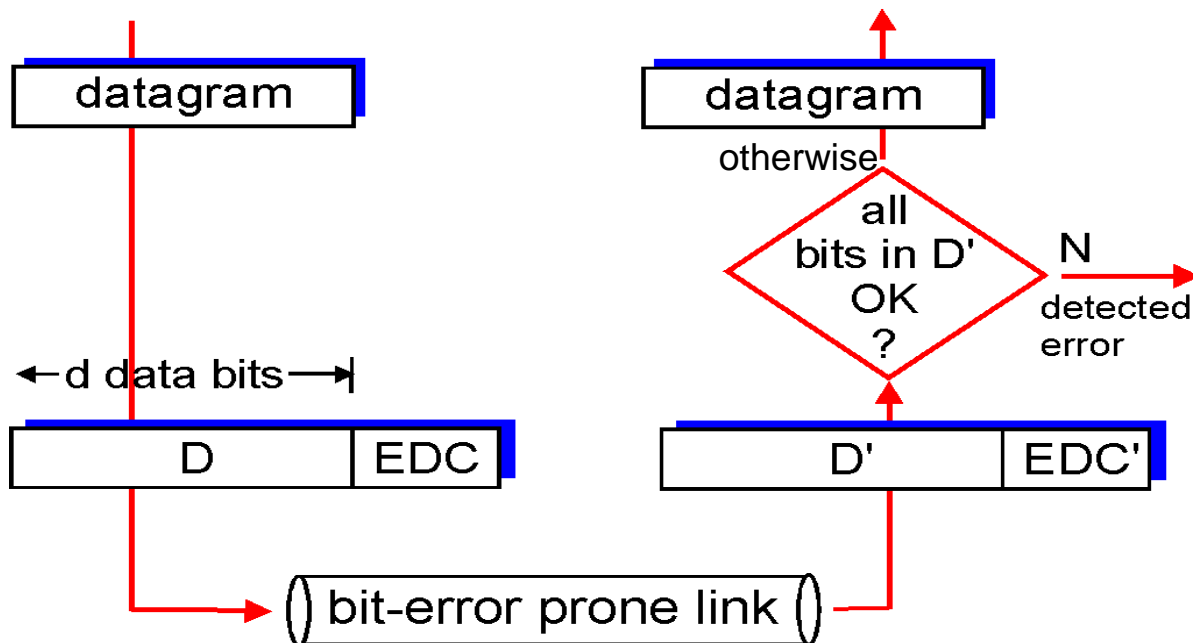
5.7 Hoạt động thường ngày - truy cập web

Phát hiện lỗi

EDC= Error Detection and Correction bits (redundancy)

D = phần dữ liệu được bảo vệ bởi trường EDC, có thể chứa các trường header

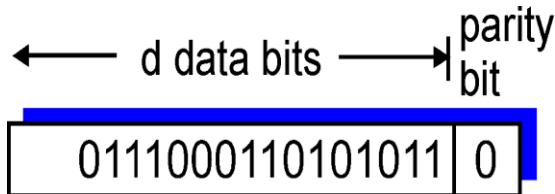
- Việc phát hiện lỗi không bảo đảm 100%!
 - giao thức có thể bỏ qua một số lỗi
 - trường EDC càng lớn thì việc phát hiện và sửa lỗi càng tốt hơn



Kiểm tra chẵn lẻ (Parity checking)

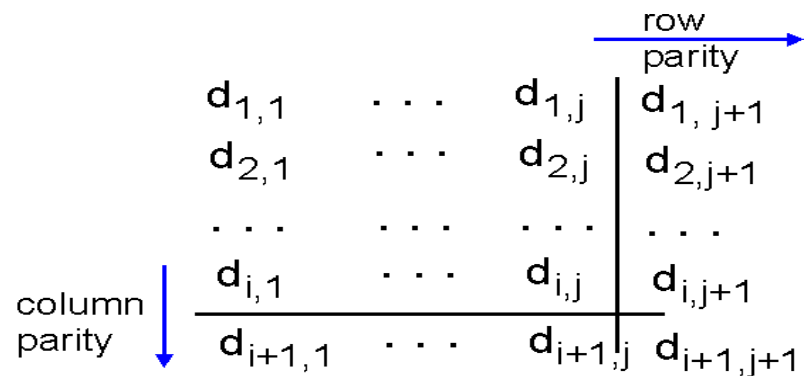
bit parity đơn:

- ❖ Phát hiện các lỗi bit đơn



bit parity 2 chiều:

- ❖ Phát hiện và sửa các lỗi bit đơn



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
<hr/>					
0	0	1	0	1	0

no errors

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
<hr/>					
0	0	1	0	1	0

parity error

*correctable
single bit error*

Internet checksum

Mục tiêu: phát hiện “các lỗi” (ví dụ, các bit bị đảo) trong packet được truyền (chú ý: chỉ được dùng tại tầng Vận chuyển)

Bên gửi:

- ❖ Xử lý các nội dung của segment như một chuỗi các số nguyên 16-bit
- ❖ checksum: là (tổng bù 1) của các nội dung của segment
- ❖ Bên gửi đặt giá trị checksum vào trong trường checksum của UDP

Bên nhận:

- ❖ Tính toán checksum của segment vừa nhận
- ❖ Kiểm tra xem giá trị của checksum vừa được tính có bằng với giá trị trong trường checksum không:
 - không – phát hiện lỗi
 - có – không phát hiện lỗi. Nhưng có thể còn có lỗi khác không?

Cyclic redundancy check

- ❖ Thuật toán phát hiện lỗi tốt hơn
- ❖ Xem đoạn bit dữ liệu, **D**, như một số nhị phân
- ❖ Chọn mẫu chiều dài $r+1$ bit (bộ khởi tạo), **G**
- ❖ Mục tiêu: chọn r bit CRC, **R**, sao cho
 - $\langle D, R \rangle$ chia hết cho G (theo cơ số 2)
 - Bên nhận biết G , chia $\langle D, R \rangle$ cho G . Nếu phần dư khác không: phát hiện lỗi!
 - Có thể phát hiện tất cả các lỗi nhỏ hơn $r+1$ bits
- ❖ Được sử dụng rộng rãi trong thực tế (Ethernet, 802.11 WiFi, ATM)

← d bits → ← r bits →



$$D * 2^r \text{ XOR } R$$

mathematical formula

CRC ví dụ

Muốn:

$$D \cdot 2^r \text{ XOR } R = nG$$

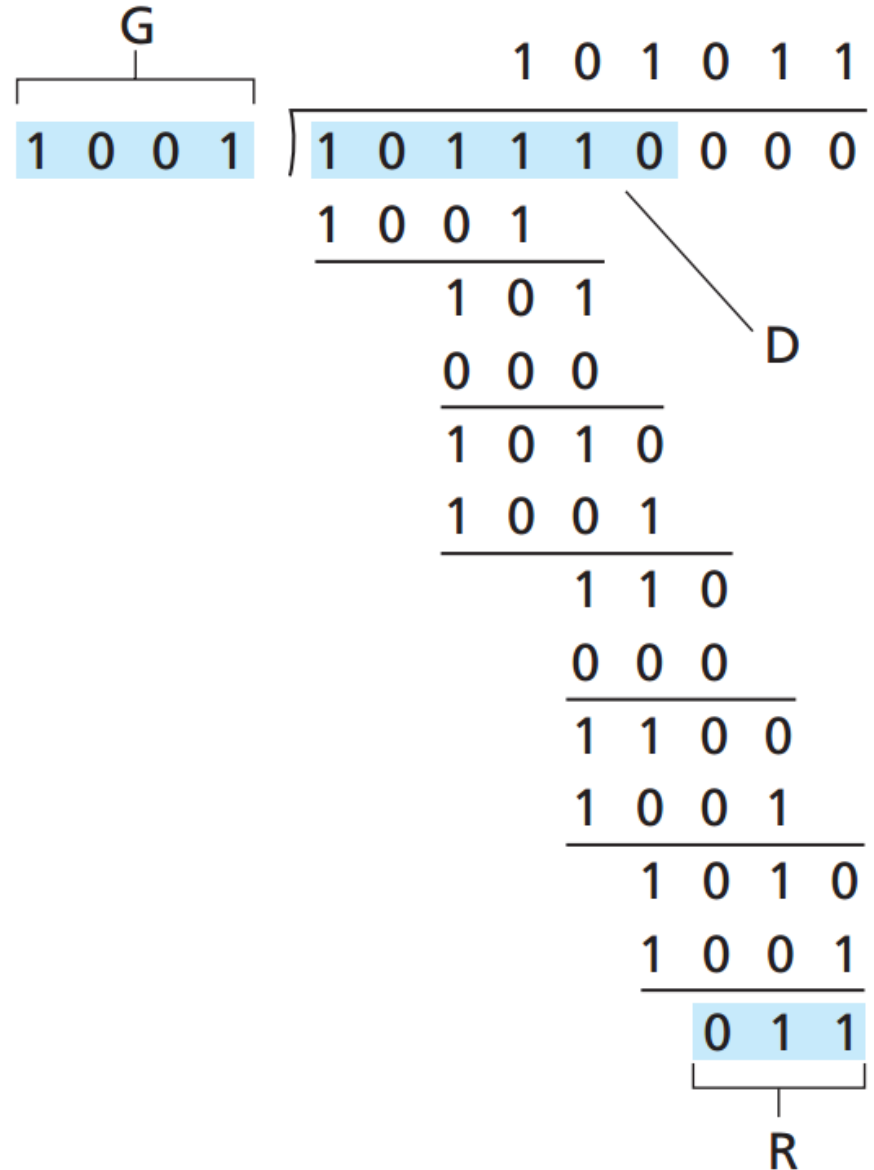
Tương đương:

$$D \cdot 2^r = nG \text{ XOR } R$$

Tương đương:

nếu chúng ta chia $D \cdot 2^r$ cho G , có được phần dư R thỏa:

$$R = \text{remainder}[\frac{D \cdot 2^r}{G}]$$



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 phát hiện lỗi và sửa lỗi

5.3 các giao thức đa truy cập

5.4 mạng LAN

- Định địa chỉ, ARP
- Ethernet
- switches
- VLANs

5.5 link virtualization: MPLS

5.6 mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Các giao thức và kết nối đa truy cập

2 kiểu “kết nối”:

❖ Điểm-điểm (point-to-point)

- PPP cho truy cập dial-up
- Kết nối point-to-point giữa Ethernet switch và host

❖ *broadcast (chia sẻ đường truyền dùng chung)*

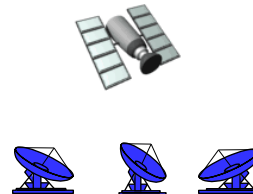
- Ethernet mô hình cũ
- upstream HFC
- 802.11 wireless LAN



shared wire (e.g.,
cabled Ethernet)



shared RF
(e.g., 802.11 WiFi)



shared RF
(satellite)



Trong buổi tiệc cocktail
(không khí và âm thanh
được chia sẻ)

Các giao thức đa truy cập

- ❖ Chia sẻ chung kênh quảng bá (broadcast) đơn
- ❖ 2 hoặc nhiều phiên truyền đồng thời bởi các node: nhiễu - giao thoa gây biến đổi mức tín hiệu
 - *collision* (đụng độ) xảy ra nếu node nhận được 2 hoặc nhiều tín hiệu tại cùng thời điểm

Giao thức đa truy cập

- ❖ Dùng thuật toán phân tán (distributed algorithm) xác định cách các node chia sẻ kênh truyền, ví dụ: xác định khi nào node có thể truyền
- ❖ Các thông báo về việc chia sẻ kênh truyền phải sử dụng chính kênh đó!
 - Không dùng thêm kênh khác để phối hợp

Giao thức đa truy cập lý tưởng

Cho trước: kênh broadcast với tốc độ R bps

Mong muốn:

1. Khi 1 node muốn truyền, nó có thể gửi dữ liệu với tốc độ R .
2. Khi M node muốn truyền, mỗi node có thể gửi với tốc độ trung bình R/M
3. Hoàn toàn phân tán:
 - không có node đặc biệt để điều phối các quá trình truyền
 - không cần đồng bộ các đồng hồ, slot
4. Đơn giản

Các giao thức MAC: phân loại

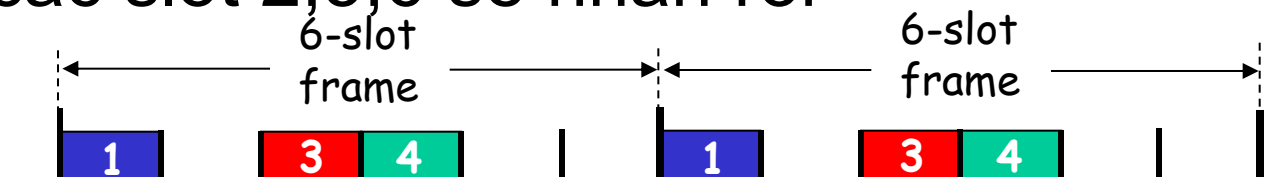
3 loại chính:

- ❖ *Phân hoạch kênh (channel partitioning)*
 - Chia kênh truyền thành “các mảnh” nhỏ hơn (các slot thời gian - TDM, tần số - FDM, mã - CDM)
 - Cấp phát mảnh này cho node để sử dụng độc quyền
- ❖ *Truy cập ngẫu nhiên (random access)*
 - Kênh truyền không được chia, cho phép đụng độ
 - “giải quyết” đụng độ
- ❖ *“Xoay vòng”*
 - Các node thay phiên nhau, nhưng nút có nhiều dữ liệu hơn được giữ thời gian truyền lâu hơn

Các giao thức MAC phân hoạch kênh: TDMA

TDMA: time division multiple access

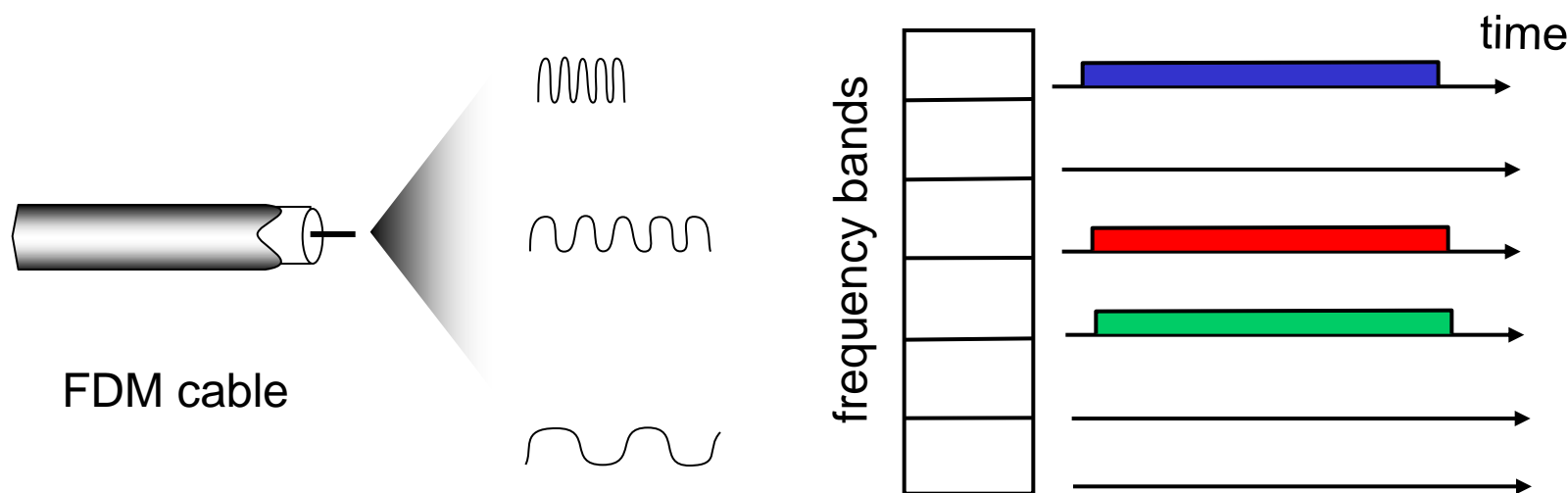
- ❖ Truy cập kênh truyền theo hình thức “xoay vòng”
- ❖ Mỗi trạm (station) có slot với độ dài cố định (độ dài = thời gian truyền packet) trong mỗi vòng (round)
- ❖ Các slot không sử dụng sẽ nhàn rỗi
- ❖ Ví dụ: LAN có 6 trạm, 1,3,4 có gói được gửi, các slot 2,5,6 sẽ nhàn rỗi



Các giao thức MAC phân hoạch kênh: FDMA

FDMA: frequency division multiple access

- ❖ Phổ kênh truyền được chia thành các dải tần số
- ❖ Mỗi trạm được gán một dải tần số cố định
- ❖ Trong thời gian không truyền, các dải tần rảnh
- ❖ Ví dụ: LAN có 6 station, 1,3,4 có packet truyền, các dải tần số 2,5,6 nhàn rỗi



Các giao thức truy cập ngẫu nhiên

- ❖ Khi 1 node có packet cần gửi
 - Truyền dữ liệu với trọn tốc độ của kênh dữ liệu R.
 - Không có sự ưu tiên giữa các node
- ❖ 2 hoặc nhiều node truyền → “đụng độ”,
- ❖ **Giao thức truy cập ngẫu nhiên MAC** xác định:
 - Cách để phát hiện đụng độ
 - Cách để giải quyết đụng độ (ví dụ: truyền lại sau đó)
- ❖ Ví dụ các giao thức MAC truy cập ngẫu nhiên:
 - slotted ALOHA
 - ALOHA
 - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA

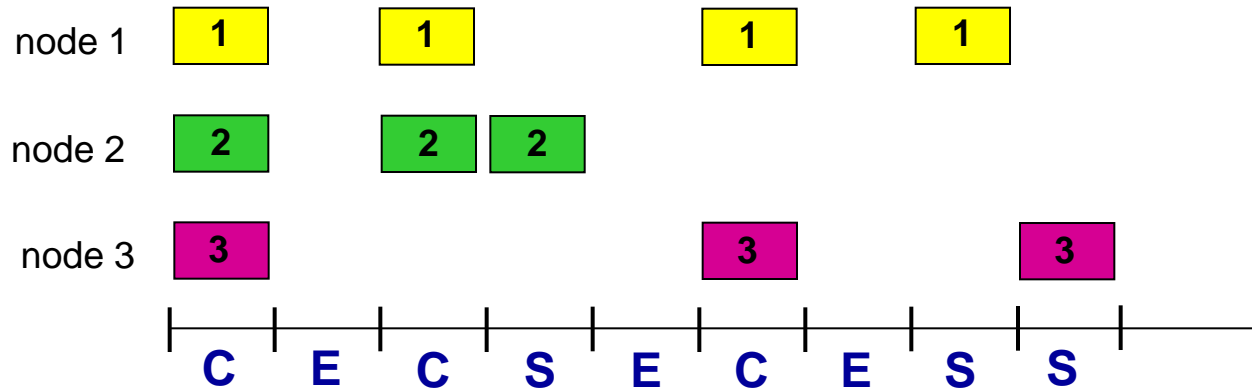
Giả thuyết:

- ❖ Tất cả các frame có cùng kích thước
- ❖ Thời gian được chia thành các slot có kích thước bằng nhau (thời gian để truyền đủ 1 frame)
- ❖ Các node bắt đầu truyền chỉ ngay tại lúc bắt đầu slot
- ❖ Các node được đồng bộ hóa
- ❖ Nếu 2 hoặc nhiều node truyền trong cùng 1 slot, thì tất cả các node đều phát hiện đụng độ

Hoạt động:

- ❖ Khi node có được frame mới, nó sẽ truyền trong slot kế tiếp
 - *Nếu không có đụng độ:* node có thể gửi frame mới trong slot kế tiếp
 - *Nếu có đụng độ:* node truyền lại frame trong mỗi slot tiếp theo với xác suất p cho đến khi thành công

Slotted ALOHA



Ưu điểm:

- ❖ Node hoạt động có thể truyền liên tục với tốc độ tối đa của kênh
- ❖ Phân tán cao: chỉ có các slot trong các node cần được đồng bộ
- ❖ Đơn giản

Nhược điểm:

- ❖ Đụng độ, lãng phí slot
- ❖ Các slot trống
- ❖ Các node có thể phát hiện tranh chấp nhanh hơn thời gian để truyền gói
- ❖ Đồng bộ hóa

Slotted ALOHA: hiệu suất

Hiệu suất: là phần slot truyền thành công trong số nhiều frame dự định truyền của nhiều node

- ❖ *Giả sử:* có N node với nhiều frame để truyền, mỗi frame truyền trong slot với xác suất là p
- ❖ Xác suất để 1 node truyền thành công trong 1 slot = $p(1-p)^{N-1}$
- ❖ Xác suất để bất kỳ node nào truyền thành công = $Np(1-p)^{N-1}$

- ❖ Hiệu suất cực đại: tìm p^* làm cực đại hóa $Np(1-p)^{N-1}$
- ❖ Với nhiều node, tìm giới hạn của $Np^*(1-p^*)^{N-1}$ khi $N \rightarrow \infty$, cho:

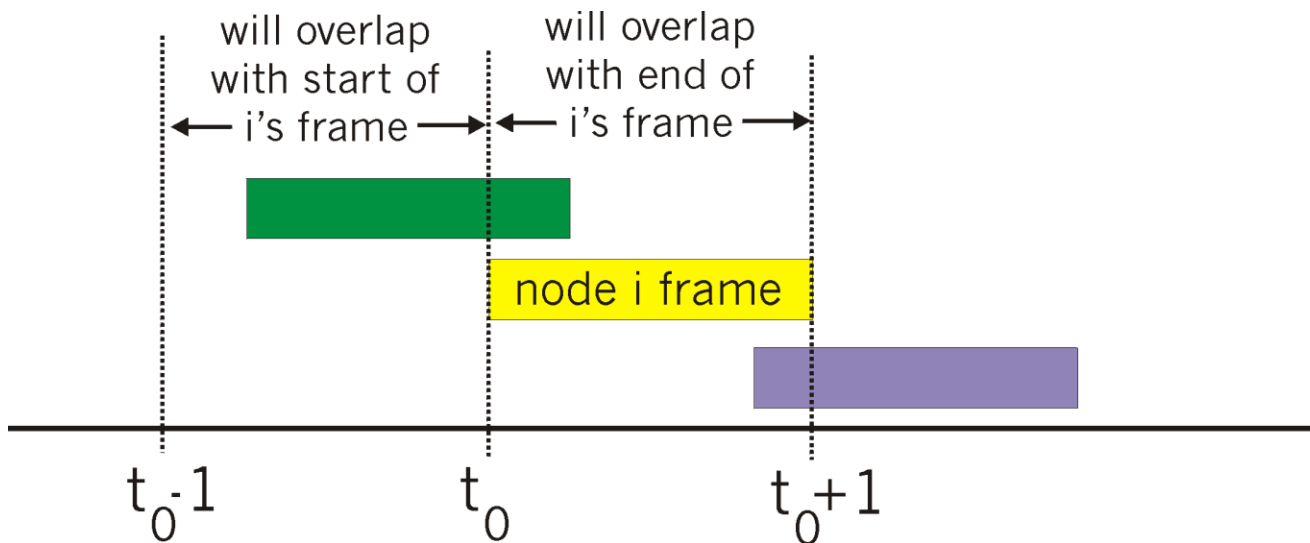
hiệu suất cực đại = $1/e = .37$

Tốt nhất: kênh hữu dụng trong khoảng 37% thời gian!



Pure (unslotted) ALOHA

- ❖ unslotted Aloha: đơn giản, không đồng bộ
- ❖ Khi frame đến
 - truyền lập tức
- ❖ Khả năng đụng độ tăng:
 - frame được truyền tại thời điểm t_0 đụng độ với các frame khác được truyền trong thời điểm $[t_0-1, t_0+1]$



Pure ALOHA: hiệu suất

$$\begin{aligned} P(\text{thành công cho node}) &= P(\text{node truyền}) \times \\ &\quad P(\text{không có node khác truyền trong } [t_0-1, t_0]) \times \\ &\quad P(\text{không có node khác truyền trong } [t_0, t_0+1]) \end{aligned}$$

$$\begin{aligned} &= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1} \\ &= p \cdot (1-p)^{2(N-1)} \end{aligned}$$

... chọn p tối ưu và sau đó cho $n \rightarrow \infty$

$$= 1/(2e) = .18$$

Thậm chí tệ hơn slotted Aloha!

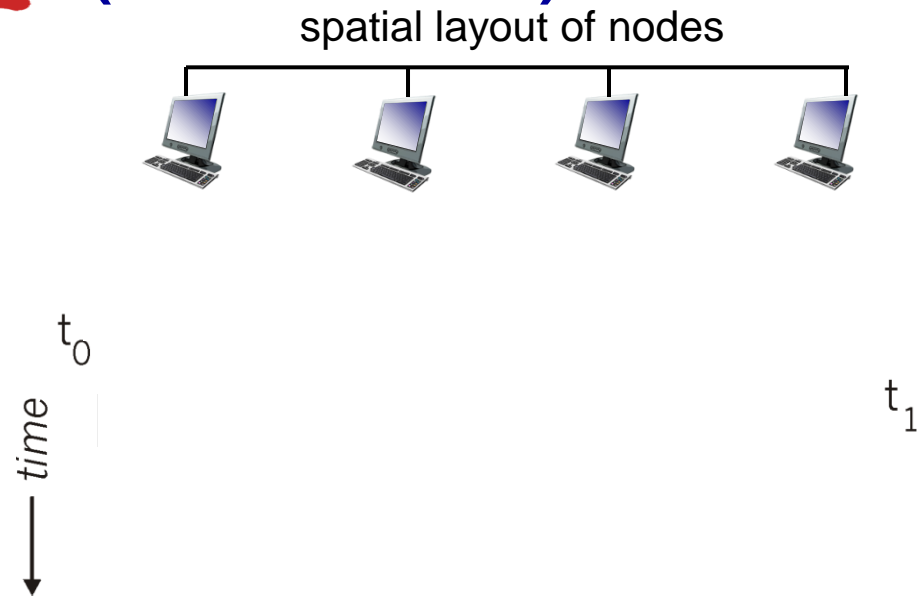
CSMA (carrier sense multiple access)

CSMA: lắng nghe trước khi truyền:

- ❖ Nếu kênh nhàn rỗi: truyền toàn bộ frame
- ❖ Nếu kênh truyền bận, trì hoãn truyền
- ❖ So sánh với con người: đừng ngắt lời người khác!

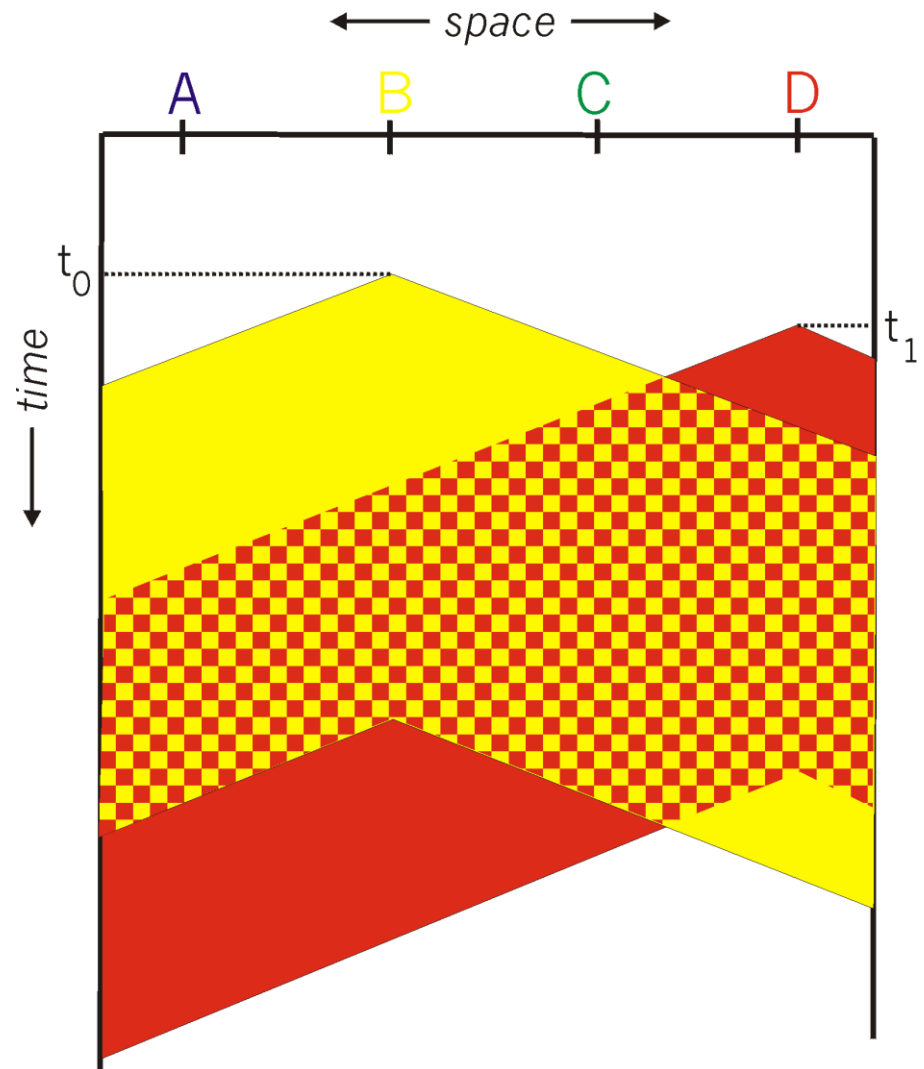
CSMA: đụng độ (collision)

- ❖ **Đụng độ có thể vẫn xảy ra:** trể lan truyền nghĩa là 2 node không thể nghe thấy quá trình truyền lẫn nhau
- ❖ **Đụng độ:** toàn bộ thời gian truyền packet bị lãng phí
 - Khoảng cách và trể lan truyền có vai trò trong việc xác định xác suất đụng độ



CSMA: đụng độ (collision)

- ❖ **Đụng độ có thể vẫn xảy ra:** trể lan truyền nghĩa là 2 node không thể nghe thấy quá trình truyền lẫn nhau
- ❖ **Đụng độ:** toàn bộ thời gian truyền packet bị lãng phí
 - Khoảng cách và trể lan truyền có vai trò trong việc xác định xác suất đụng độ



CSMA/CD (collision detection)

CSMA/CD: trì hoãn như trong CSMA

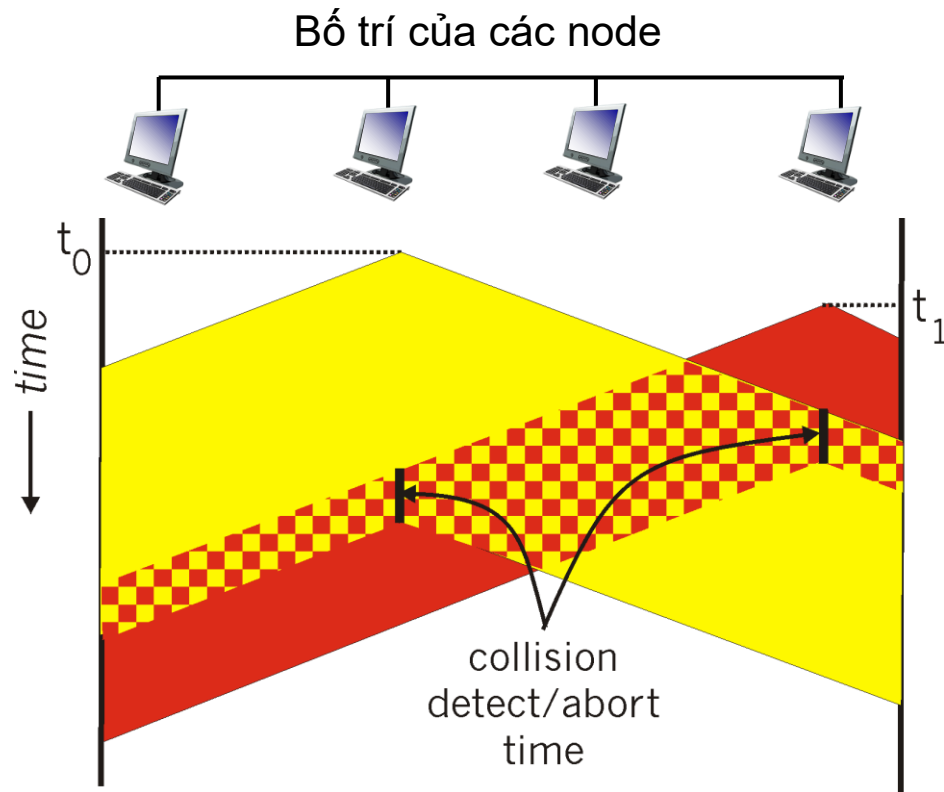
- Đụng độ được phát hiện trong thời gian ngắn
- Thông tin đang truyền bị hủy bỏ, giảm sự lãng phí kênh.

❖ Phát hiện đụng độ:

- Dễ dàng trong các mạng LAN hữu tuyến: đo cường độ tín hiệu, so sánh với các tín hiệu đã được truyền và nhận
- Khó thực hiện trong mạng LAN vô tuyến: cường độ truyền cục bộ lấn át cường độ tín hiệu nhận

❖ Tương tự như hành vi của con người: đàm thoại lịch sự

CSMA/CD (collision detection)



Thuật toán Ethernet CSMA/CD

1. NIC nhận datagram từ tầng network, tạo frame
2. Nếu NIC dò được kênh rồi, nó sẽ bắt đầu việc truyền frame. Nếu NIC dò được kênh bận, đợi cho đến khi kênh rảnh, sau đó mới truyền.
3. Nếu NIC truyền toàn bộ frame mà không phát hiện việc truyền khác, NIC được truyền toàn bộ frame đó!
4. Nếu NIC phát hiện có phiên truyền khác trong khi đang truyền, thì nó sẽ hủy bỏ việc truyền và phát tín hiệu tắc nghẽn
5. Sau khi hủy bỏ truyền, NIC thực hiện *binary (exponential) backoff*:
 - Sau lần đụng độ thứ m , NIC chọn ngẫu nhiên số K trong khoảng $\{0, 1, 2, \dots, 2^m - 1\}$. NIC sẽ đợi $K \cdot 512$ thời gian truyền bit (bit time), sau đó trở lại bước 2
 - Càng nhiều đụng độ thì sẽ có khoảng thời gian chờ dài hơn

CSMA/CD hiệu suất

- ❖ T_{prop} = độ trễ lan truyền lớn nhất (max prop delay) giữa 2 node trong mạng LAN
- ❖ t_{trans} = thời gian để truyền frame có kích thước lớn nhất

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- ❖ Hiệu suất tiến tới 1
 - khi t_{prop} tiến tới 0
 - khi t_{trans} tiến tới vô cùng
- ❖ Hiệu suất tốt hơn ALOHA: đơn giản, chi phí thấp và điều khiển phân tán!

Các giao thức MAC “ Xoay vòng ”

Các giao thức phân hoạch kênh MAC (channel partitioning MAC protocols):

- Chia sẻ kênh hiệu quả và công bằng với tải lớn
- Không hiệu quả ở tải thấp: trễ khi truy cập kênh, 1 node chỉ được cấp phát $1/N$ bandwidth ngay cả khi chỉ có 1 node hoạt động!

Các giao thức MAC truy cập ngẫu nhiên (random access MAC protocols)

- Hiệu quả tại tải thấp: node đơn có thể dùng hết khả năng của kênh
- Tải cao: độn độ cao

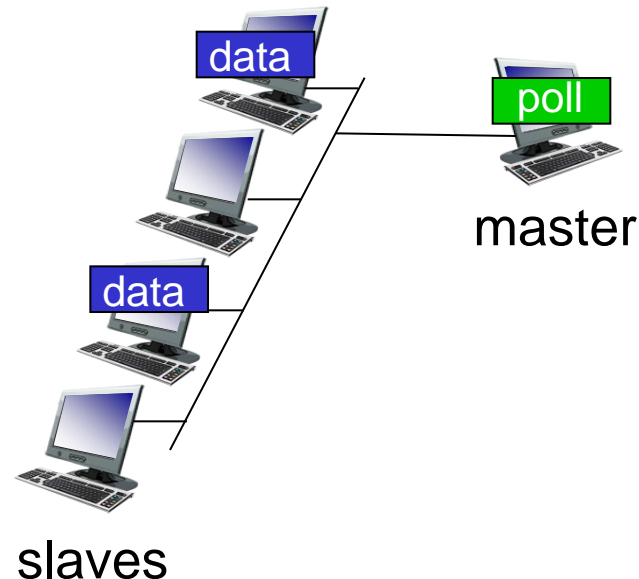
Các giao thức “ Xoay vòng ” (“taking turns” protocols)

Tìm kiếm giải pháp dung hòa tốt nhất!

Các giao thức MAC “Xoay vòng”

polling:

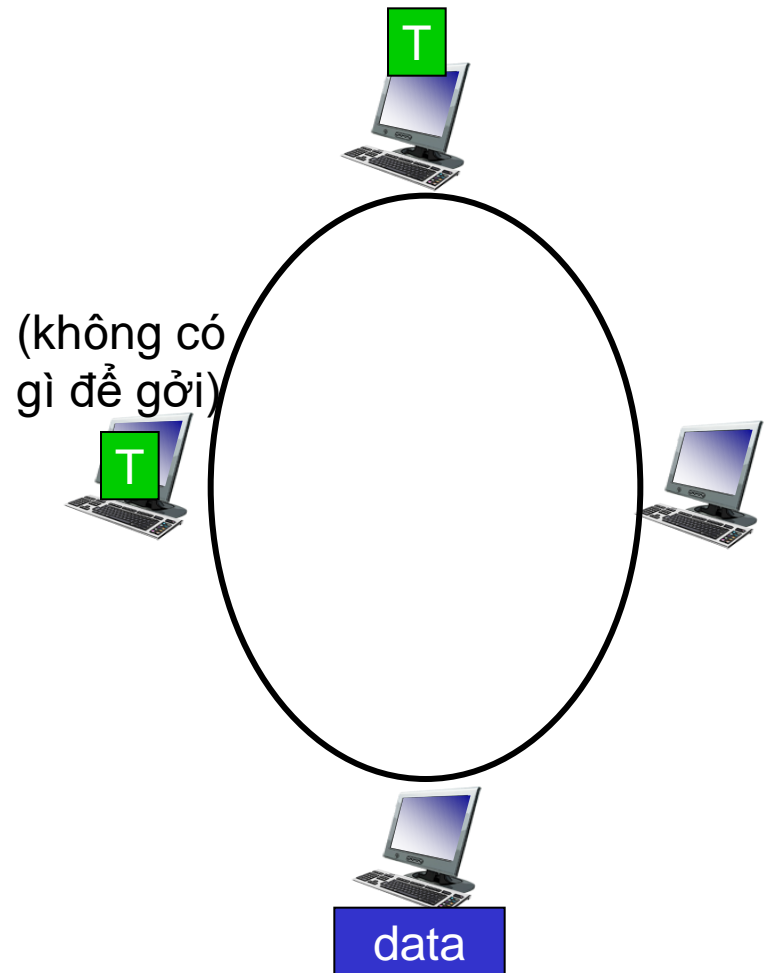
- ❖ Node chủ (master node) “mời” các node con (slave node) truyền lần lượt
- ❖ Thường được sử dụng với các thiết bị con “không thông minh”
- ❖ Quan tâm:
 - Chi phí cho việc điều phối (polling overhead)
 - Độ trễ (latency)
 - Có 1 điểm chịu lỗi (master)



Các giao thức MAC “Xoay vòng”

Chuyển token:

- ❖ Điều hành việc chuyển *token* tuần tự từ 1 node đến node kế tiếp.
- ❖ Gói token
- ❖ Quan tâm:
 - Chi phí cho việc chuyển token (token overhead)
 - Độ trễ (latency)
 - Có 1 điểm chịu lỗi (token)



Tổng kết các giao thức MAC

- ❖ *Phân hoạch kênh*, theo thời gian, tần số hoặc mã
 - Phân chia theo thời gian (Time Division), phân chia theo tần số (Frequency Division)
- ❖ *Truy cập ngẫu nhiên* (động),
 - ALOHA, S-ALOHA, CSMA, CSMA/CD
 - Cảm nhận sóng mang (carrier sensing): dễ dàng trong một số kỹ thuật (có dây), khó thực hiện trong các công nghệ khác (không dây)
 - CSMA/CD được dùng trong Ethernet
 - CSMA/CA được dùng trong 802.11
- ❖ *Xoay vòng*
 - Điều phối từ đơn vị trung tâm, truyền token
 - bluetooth, FDDI, token ring

Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

5.5 Link virtualization: MPLS

5.6 Mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Địa chỉ MAC và ARP

❖ Địa chỉ IP 32-bit:

- *Địa chỉ tầng network* cho interface
- Được sử dụng trong chức năng chuyển dữ liệu tầng 3 (tầng network)

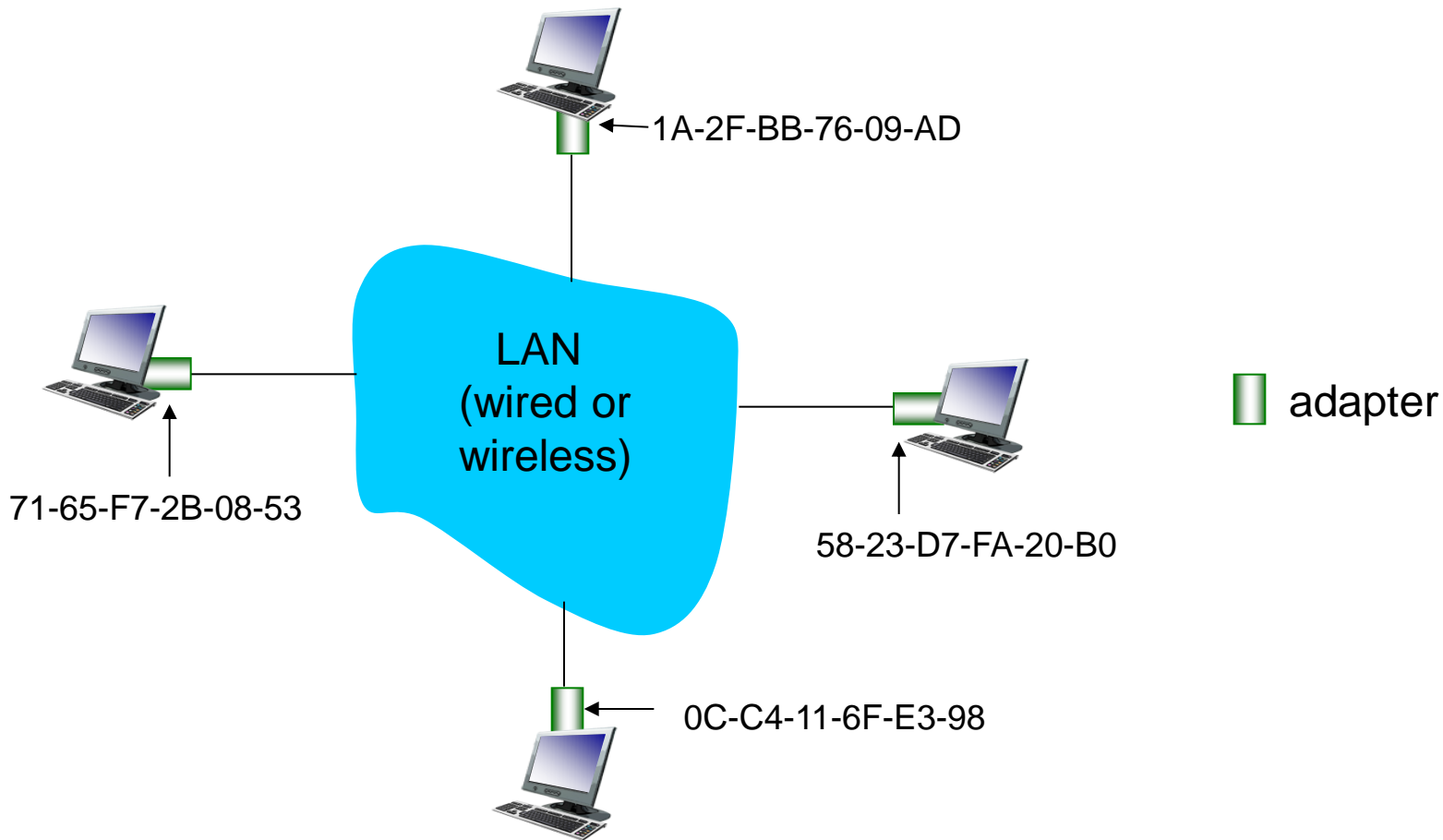
❖ Địa chỉ MAC (hoặc địa chỉ LAN hoặc physical hoặc Ethernet) :

- Chức năng: được sử dụng “cục bộ” để chuyển frame từ 1 interface này đến 1 interface được kết nối vật lý trực tiếp với nhau (cùng mạng, trong ngữ cảnh vùng địa chỉ IP)
- Địa chỉ MAC 48 bit (cho hầu hết các mạng LAN) được ghi vào trong NIC ROM, hoặc thiết lập trong phần mềm
- Ví dụ: 1A-2F-BB-76-09-AD

Ghi dưới dạng số thập lục phân (hệ 16)
(mỗi “số” đại diện 4 bit)

Địa chỉ MAC và ARP

Mỗi adapter trên mạng LAN có địa chỉ *LAN* duy nhất

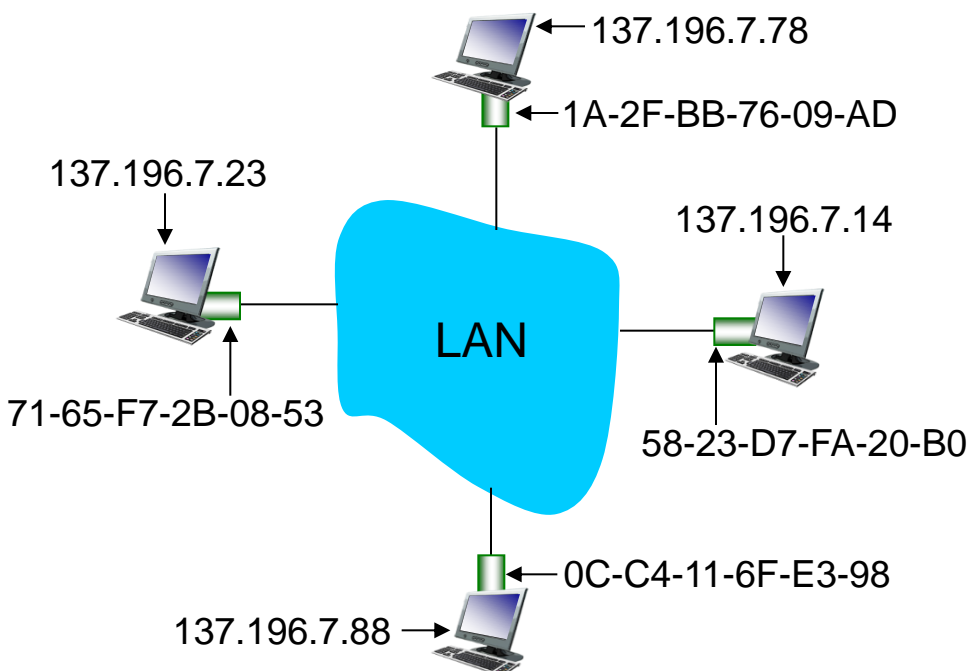


Địa chỉ LAN(tt)

- ❖ Sự phân bổ địa chỉ MAC được quản lý bởi IEEE
- ❖ Nhà sản xuất mua phần không gian địa chỉ MAC (bảo đảm duy nhất)
- ❖ So sánh:
 - Địa chỉ MAC: như là số chứng minh nhân dân
 - Địa chỉ IP: như là địa chỉ bưu điện
- ❖ Địa chỉ MAC không phân cấp, có tính di chuyển
 - Có thể di chuyển card LAN từ 1 mạng LAN này tới mạng LAN khác
- ❖ Địa chỉ IP phân cấp, không di chuyển được
 - Địa chỉ phụ thuộc vào subnet IP mà node đó gắn vào

ARP: address resolution protocol

Hỏi: làm cách nào để xác định địa chỉ MAC của interface khi biết được địa chỉ IP của nó?



Bảng ARP: mỗi node (host, router) trên mạng LAN có bảng ARP

- Địa chỉ IP/MAC ánh xạ cho các node trong mạng LAN:

< địa chỉ IP; địa chỉ MAC; TTL >

- TTL (Time To Live): thời gian sau đó địa chỉ ánh xạ sẽ bị lãng quên (thông thường là 20 phút)

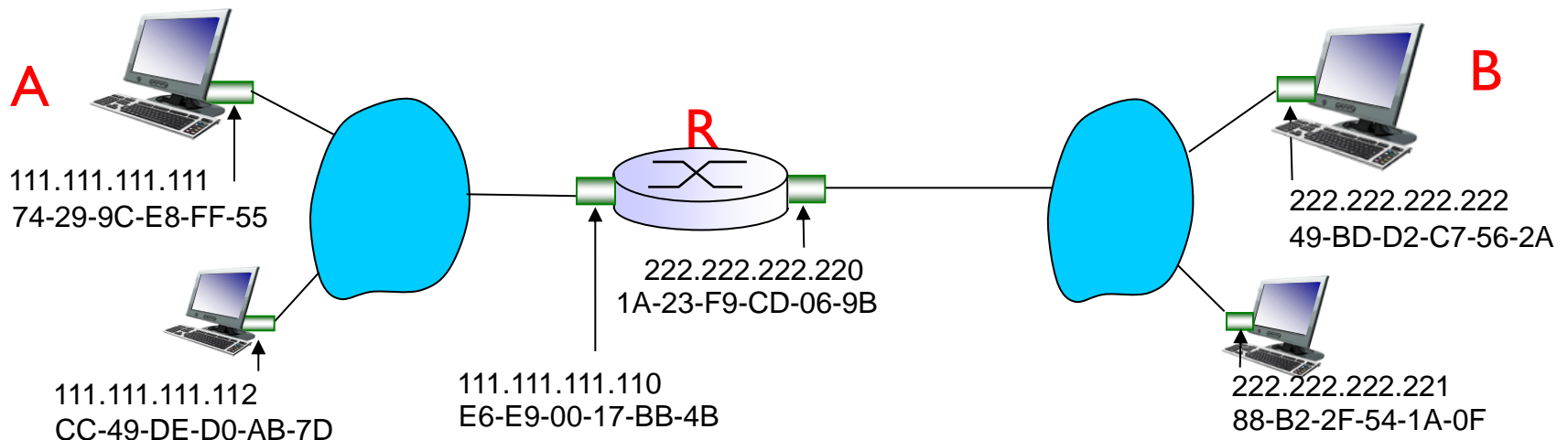
Giao thức ARP: cùng mạng LAN

- ❖ A muốn gửi datagram tới B
 - Địa chỉ MAC của B không có trong bảng ARP của A.
- ❖ A sẽ gửi **quảng bá (broadcasts)** gói tin ARP query có chứa địa chỉ IP của B
 - Địa chỉ MAC đích = FF-FF-FF-FF-FF-FF
 - Tất cả các node trên mạng LAN sẽ nhận ARP query này
- ❖ B nhận gói tin ARP, trả lời tới A với địa chỉ MAC của B
 - Frame được gửi tới địa chỉ MAC của A (unicast)
- ❖ A sẽ lưu lại cặp địa chỉ IP-MAC trong bảng ARP của nó cho tới khi thông tin này hết hạn sử dụng
 - soft state: thông tin hết hạn (bỏ đi) trừ khi được làm mới
- ❖ ARP là giao thức “plug-and-play”:
 - Các nodes tạo bảng ARP của nó không cần sự can thiệp của người quản trị mạng

Addressing: định tuyến tới mạng LAN khác

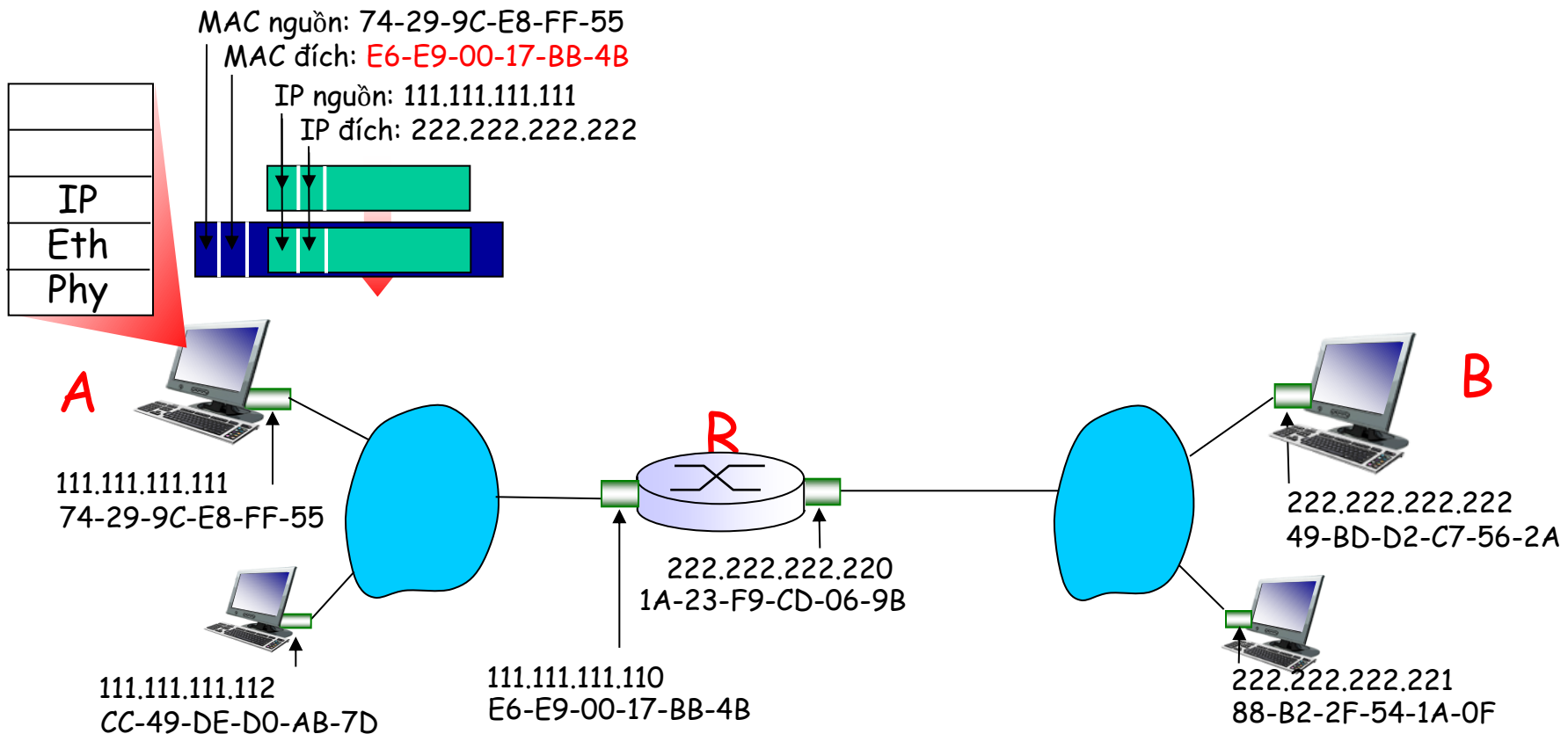
Từng bước: **gửi datagram từ A tới B thông qua R**

- tập trung vào gán địa chỉ – tại tầng IP (datagram) và MAC (frame)
- giả sử A biết địa chỉ IP của B
- giả sử A biết địa chỉ IP của router gateway R (cách nào?)
- giả sử A biết địa chỉ MAC của R (cách nào?)



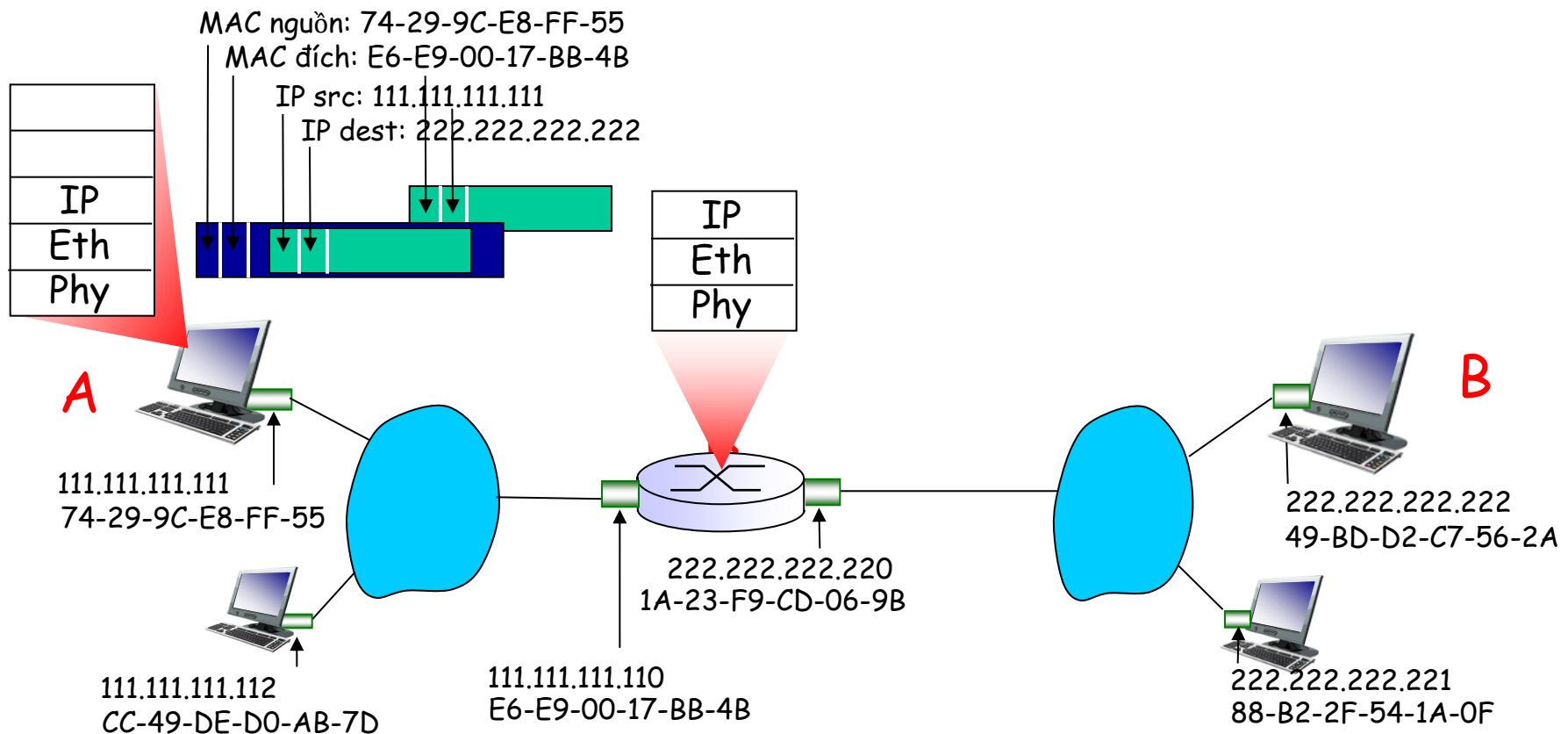
Addressing: định tuyến tới mạng LAN khác

- ❖ A tạo IP datagram với IP nguồn A, đích B
- ❖ A tạo frame tầng Liên kết dữ liệu với địa chỉ MAC của R là địa chỉ đích, frame này chứa IP datagram từ A tới B



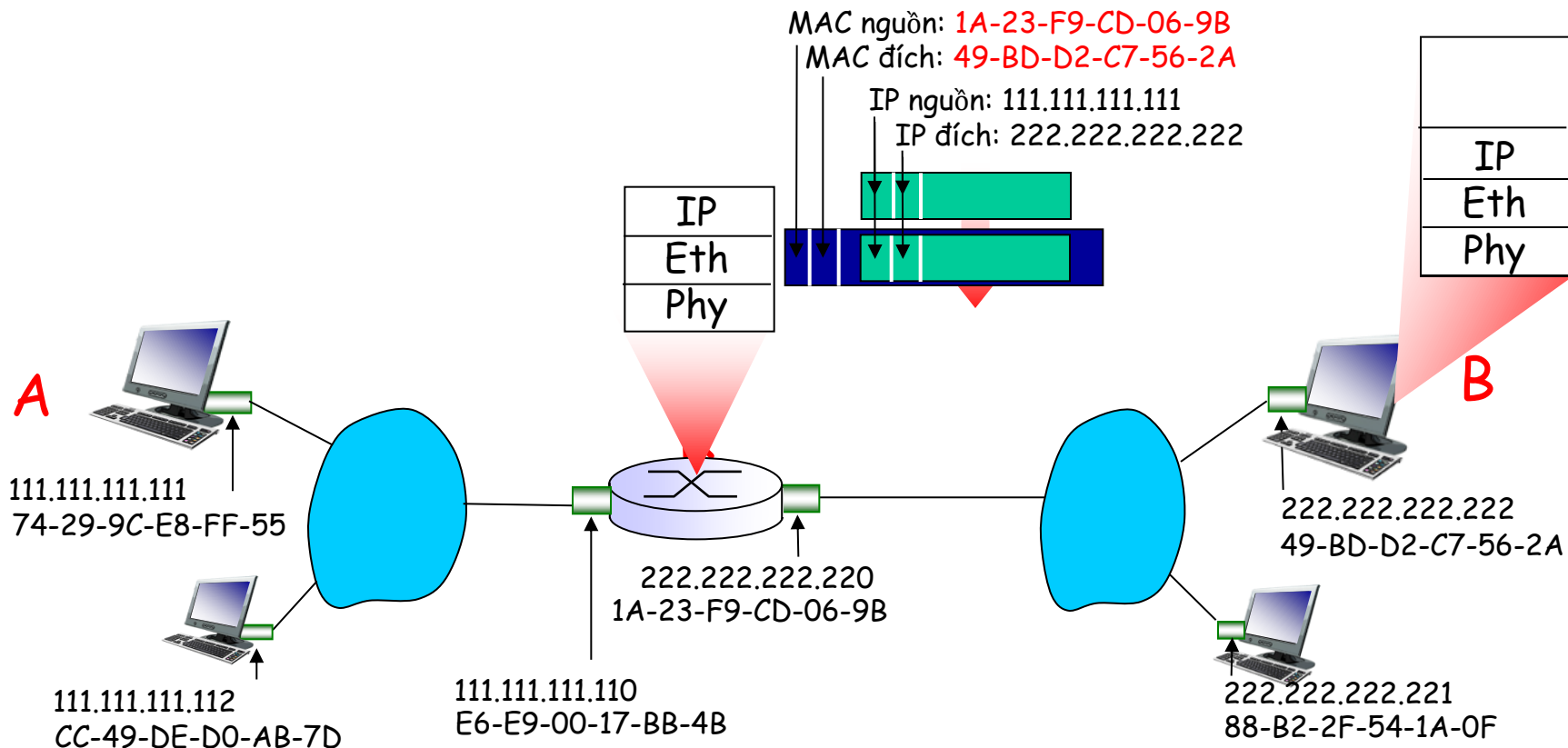
Addressing: định tuyến tới mạng LAN khác

- ❖ frame được gửi từ A tới R
- ❖ frame được R nhận, datagram được gỡ bỏ, được chuyển tới IP



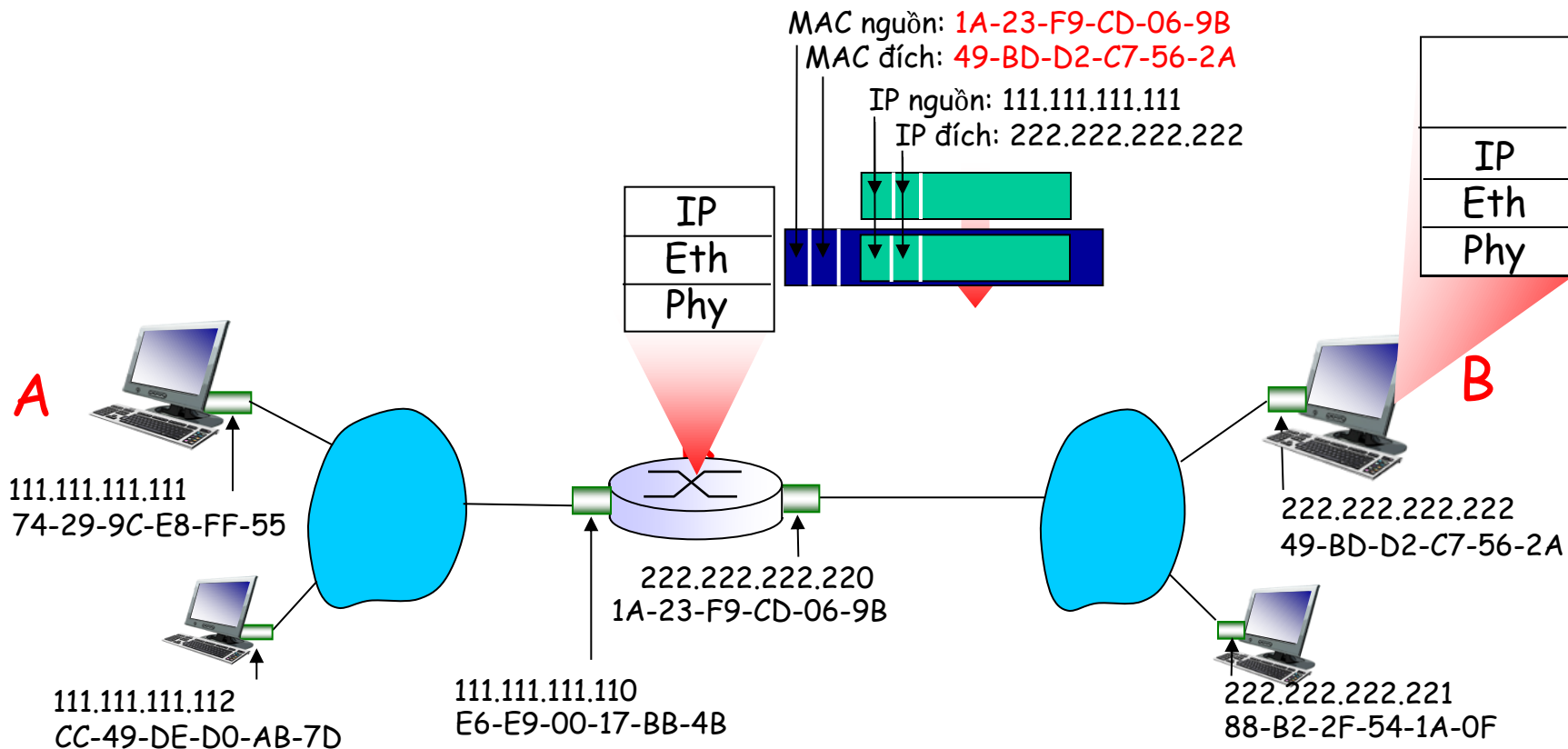
Addressing: định tuyến tới mạng LAN khác

- ❖ R sẽ chuyển tiếp datagram với IP nguồn A, đích B
- ❖ R tạo frame tầng Liên kết dữ liệu với địa chỉ MAC của B là địa chỉ đích, frame này chứa IP datagram từ A tới B



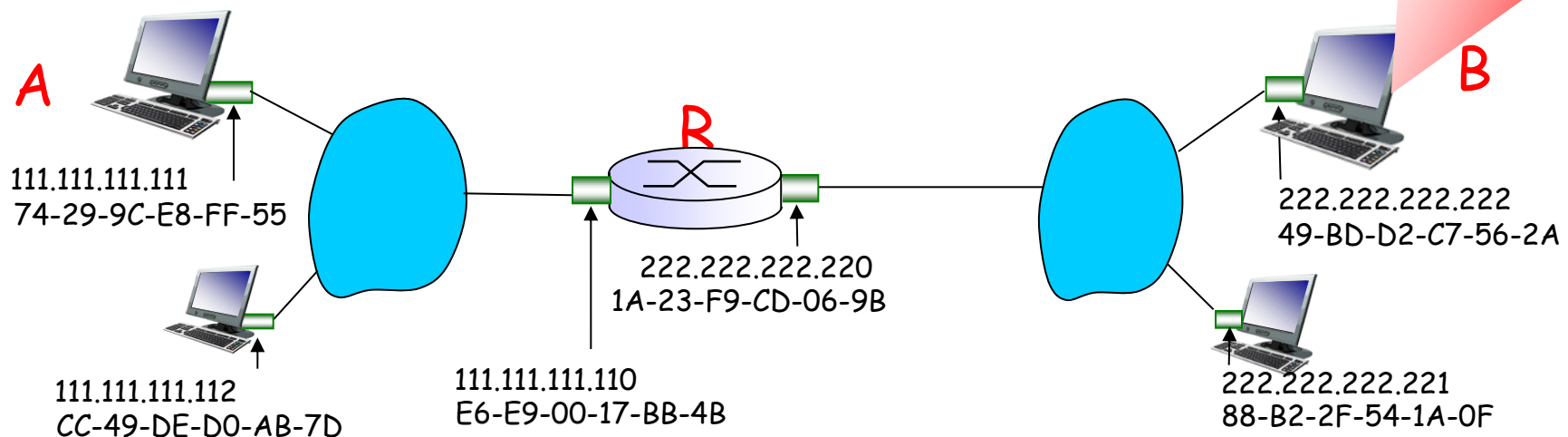
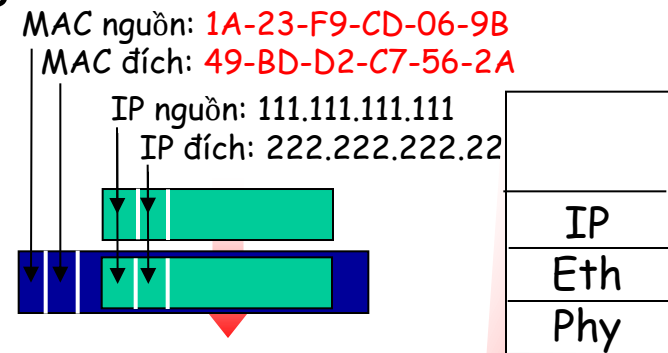
Addressing: định tuyến tới mạng LAN khác

- ❖ R chuyển tiếp datagram với IP nguồn A, đích B
- ❖ R tạo frame tầng Liên kết dữ liệu với địa chỉ MAC của B là địa chỉ đích, frame này chứa IP datagram từ A-tới-B



Addressing: định tuyến tới mạng LAN khác

- ❖ R chuyển tiếp datagram với IP nguồn A, đích B
- ❖ R tạo frame tầng Liên kết dữ liệu với địa chỉ MAC của B là địa chỉ đích, frame chứa IP datagram từ A-tới-B



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

5.5 Link virtualization: MPLS

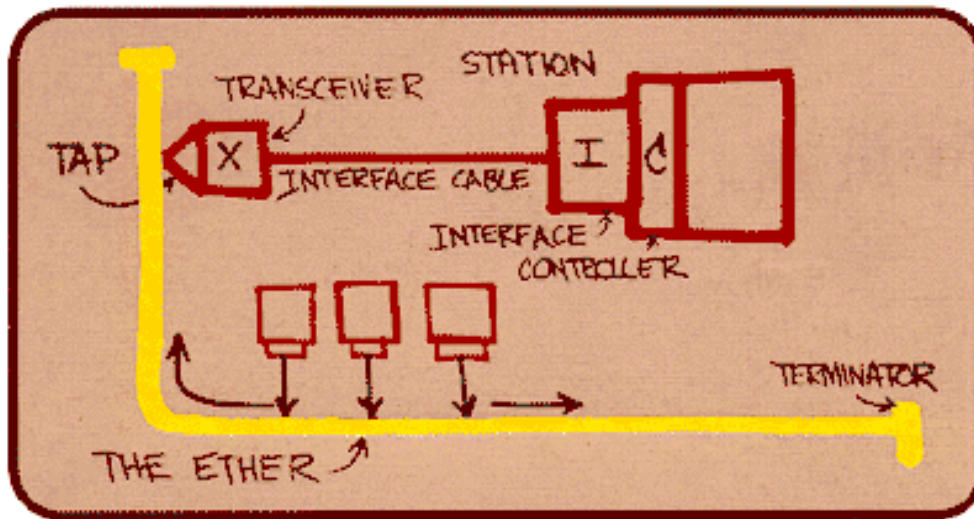
5.6 Mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Ethernet

Công nghệ mạng LAN hữu tuyến “chiếm ưu thế”:

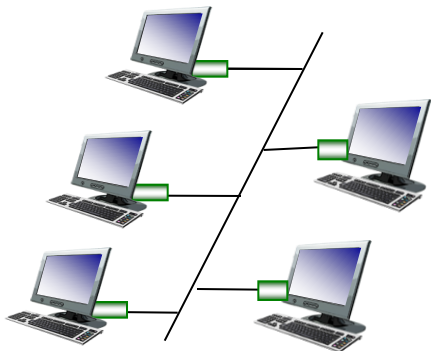
- ❖ \$20 cho NIC
- ❖ Công nghệ mạng LAN đầu tiên được sử dụng rộng rãi
- ❖ Đơn giản hơn, rẻ hơn mạng LAN token và ATM
- ❖ Tăng tốc độ trung bình từ: 10 Mbps – 10 Gbps



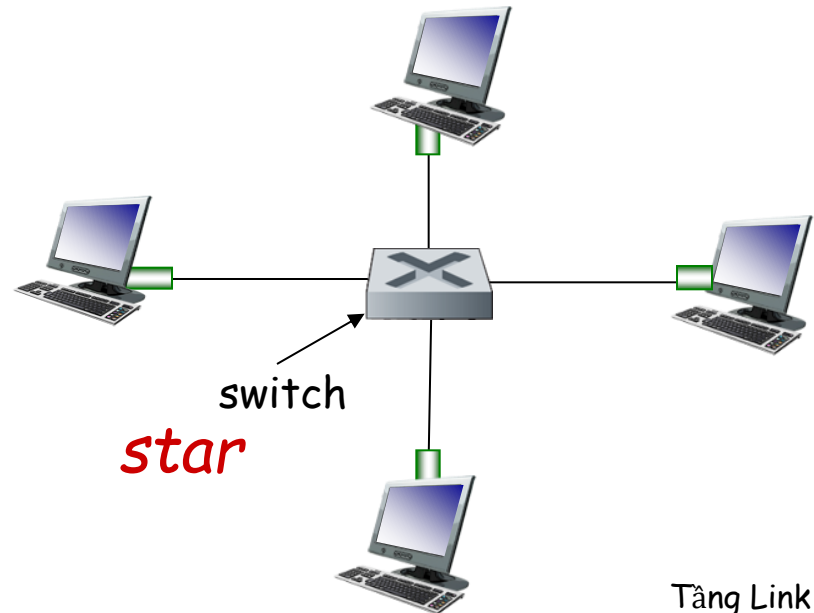
Phác thảo Ethernet của Metcalfe

Ethernet: cấu trúc vật lý

- ❖ **bus**: phổ biến trong giữa thập niên 90
 - Tất cả các node trong cùng vùng xung đột (collision domain) (có thể đụng độ lẫn nhau)
- ❖ **star**: chiếm ưu thế ngày nay
 - **switch** hoạt động ở trung tâm
 - Mỗi chặng kết nối Ethernet hoạt động riêng biệt (các node không đụng độ lẫn nhau)



bus: cáp đồng trục



Cấu trúc frame Ethernet

Adapter bên gửi sẽ đóng gói IP datagram (hoặc gói thuộc giao thức khác của tầng mạng) trong **Ethernet frame**



Preamble:

- ❖ 7 byte đầu, mỗi byte có giá trị 10101010; byte cuối có giá trị 10101011
- ❖ Được sử dụng để đồng bộ tốc độ đồng hồ của bên gửi và nhận

Cấu trúc frame Ethernet (tt)

- ❖ **Addresses:** 6 byte địa chỉ MAC nguồn, đích
 - Nếu adapter nhận frame với địa chỉ đích đúng là của nó, hoặc với địa chỉ broadcast (như là ARP packet), thì nó sẽ chuyển dữ liệu trong frame tới giao thức tầng Mạng
 - Ngược lại, adapter sẽ hủy frame
- ❖ **Type:** chỉ ra giao thức tầng cao hơn (thường là IP nhưng cũng có thể là những loại khác như là Novell IPX, AppleTalk)
- ❖ **CRC:** kiểm tra lỗi tại bên nhận
 - Lỗi được phát hiện: frame bị bỏ

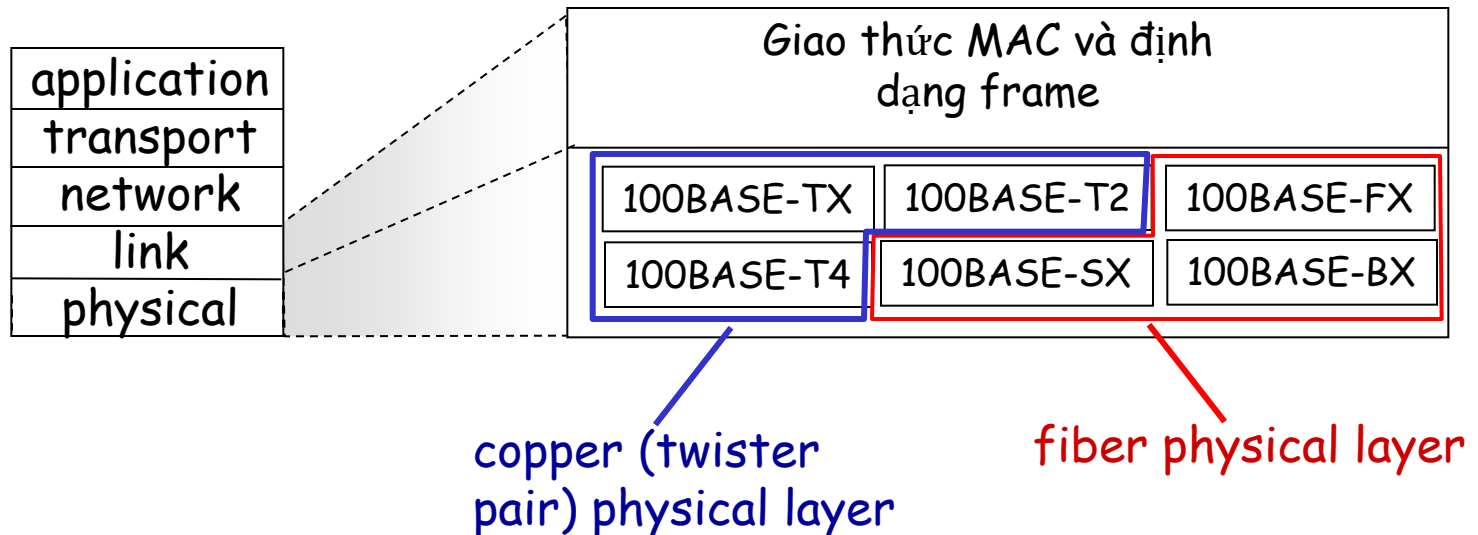


Ethernet: không tin cậy, không kết nối

- ❖ *Connectionless (không kết nối)*: không bắt tay giữa các NIC gửi và nhận
- ❖ *Unreliable (không tin cậy)*: NIC nhận sẽ không gửi thông báo nhận thành công (acks) hoặc không thành công (nacks) đến các NIC gửi
 - Dữ liệu trong các frame bị bỏ sẽ được khôi phục lại nếu bên gửi dùng dịch vụ tin cậy của tầng cao hơn (như là TCP) còn không thì dữ liệu đã bị bỏ sẽ mất luôn
- ❖ Giao thức MAC của Ethernet: unslotted *CSMA/CD với binary backoff*

Chuẩn Ethernet 802.3: Tầng Liên kết dữ liệu & Tầng Vật lý

- ❖ **Nhiều** chuẩn Ethernet khác nhau
 - Cùng giao thức MAC và định dạng frame
 - Tốc độ khác nhau: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
 - Môi trường truyền tầng Vật lý khác nhau: fiber, cable



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

5.5 Link virtualization: MPLS

5.6 Mạng trung tâm dữ liệu

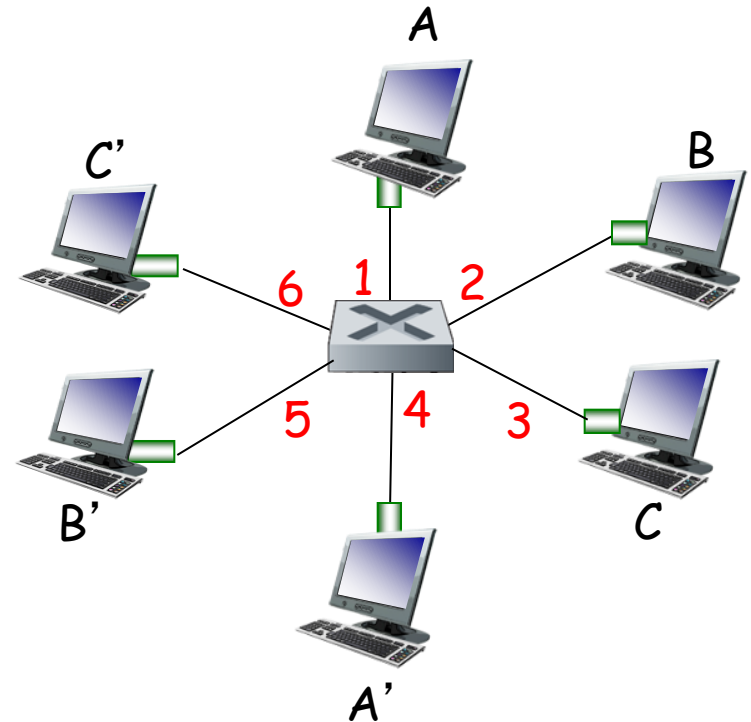
5.7 Hoạt động thường ngày - truy cập web

Ethernet switch

- ❖ Thiết bị tầng Liên kết dữ liệu: đóng vai trò tích cực
 - Lưu (store) và chuyển tiếp (forward) các frame Ethernet
 - Xem xét địa chỉ MAC của frame đến, **chọn lựa** chuyển tiếp frame tới 1 hay nhiều đường link đi ra khi frame được chuyển tiếp vào từng chặng mạng, dùng CSMA/CD để truy nhập chặng mạng
- ❖ **Transparent (trong suốt)**
 - Các host không phát hiện được sự hiện diện của các switch
- ❖ **plug-and-play, tự học**
 - Các switch không cần được cấu hình

Switch: nhiều phiên truyền đồng thời

- ❖ Các host kết nối trực tiếp tới switch
- ❖ Switch lưu tạm (buffer) các packet
- ❖ Giao thức Ethernet được sử dụng trên *mỗi* đường kết nối vào, nhưng không có đụng độ; full duplex
 - Mỗi đường kết nối là 1 miền đụng độ (collision domain) của riêng nó
- ❖ **switching**: A-tới-A' và B-tới-B' có thể truyền đồng thời mà không có đụng độ xảy ra



switch với 6 interface
(1,2,3,4,5,6)

Bảng switch forwarding

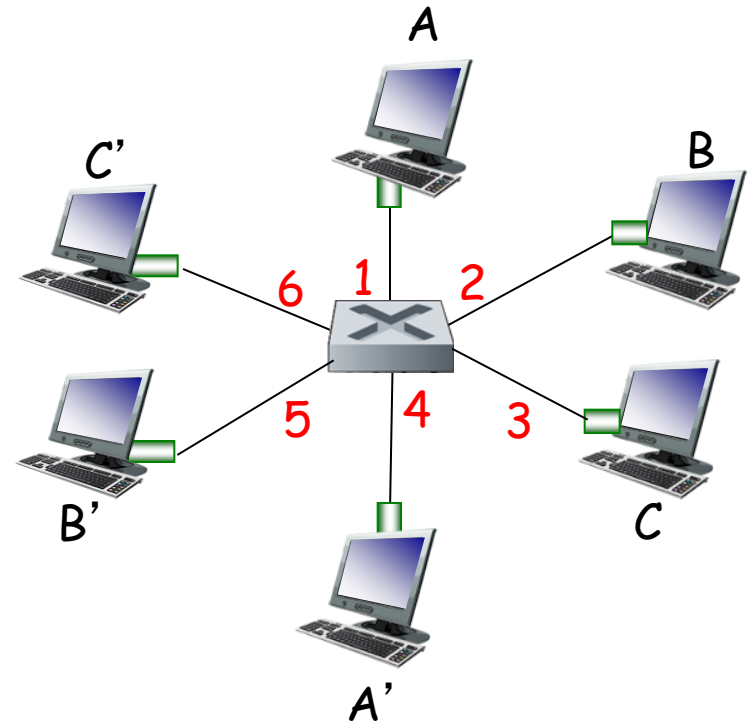
Hỏi: làm thế nào để switch biết tới A' thì sẽ thông qua interface 4 và tới B' thì thông interface 5?

❖ Đáp: mỗi switch có **một bảng switch**, mỗi dòng gồm:

- (địa chỉ MAC của host, interface để tới được host đó, time stamp)
- Giống như bảng định tuyến!

Q: những dòng được tạo và được duy trì như thế nào trong bảng switch?

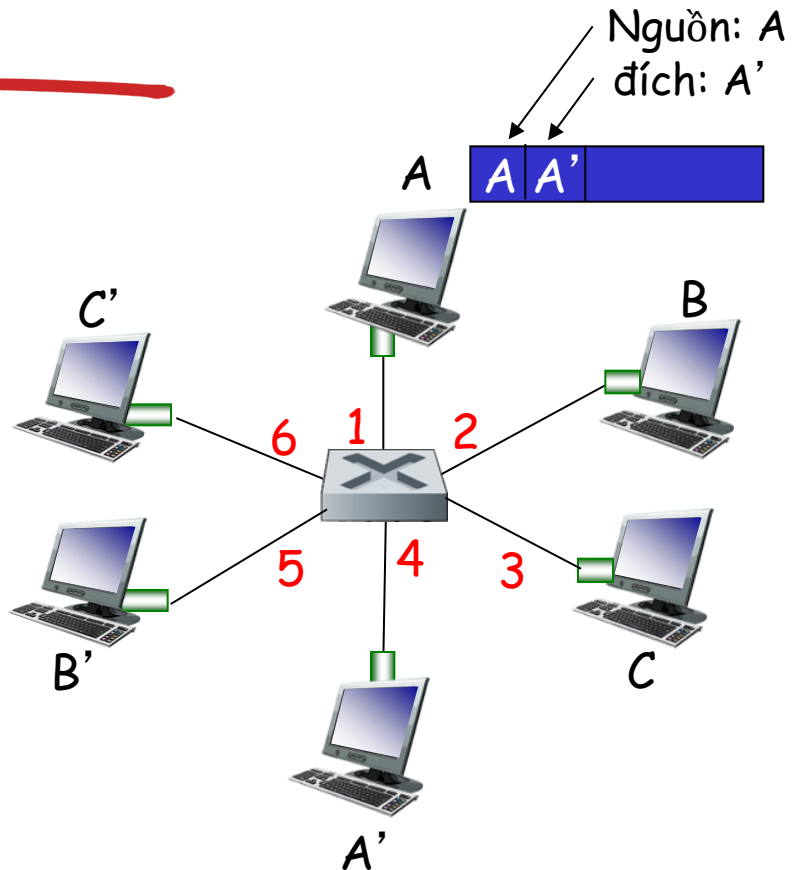
- Có giống như giao thức định tuyến hay không?



switch với 6 interface
(1,2,3,4,5,6)

Switch: tự học

- ❖ Switch *học* về vị trí các host có thể truyền tới được thông qua các interface kết nối với các host đó
 - Khi switch nhận được frame, switch “học” vị trí của bên gửi: cổng vào - incoming LAN segment
 - Ghi lại cặp thông tin (bên gửi-vị trí) vào trong bảng switch



MAC addr	interface	TTL
A	1	60

*Bảng Switch
(ban đầu trống)*

Switch: lọc/chuyển tiếp frame

Khi switch nhận được frame:

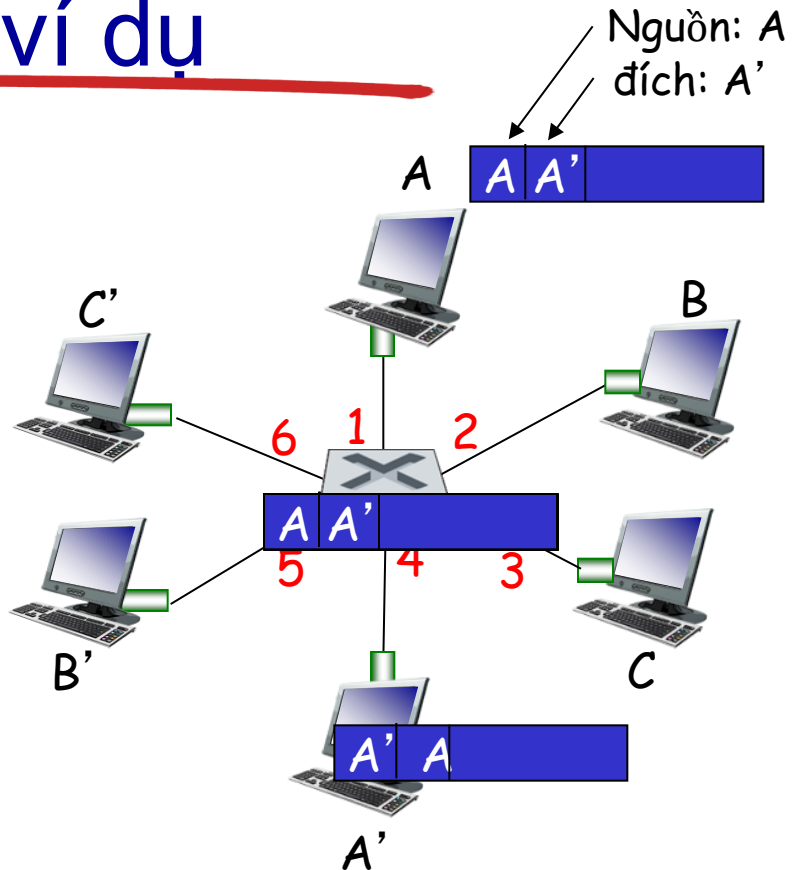
1. Ghi lại cổng kết nối vào, địa chỉ MAC của host gửi
2. Tạo chỉ mục bảng switch bằng địa chỉ MAC đích
3. Nếu tìm thấy thông tin đích đến
thì {
 nếu đích đến nằm trên phân đoạn mạng gửi frame
 đến
 thì bỏ frame
 ngược lại chuyển tiếp frame trên interface được
 chỉ định bởi thông tin trong bảng switch
 }
 ngược lại flood /* chuyển tiếp trên tất cả interface
 ngoại trừ interface nhận frame đó*/

Tự học, chuyển tiếp: ví dụ

❖ frame có đích đến là A', vị trí của A' không biết: *flood*

❖ Đích A có vị trí đã được biết trước:

gửi chọn lọc chỉ trên 1 đường kết nối duy nhất

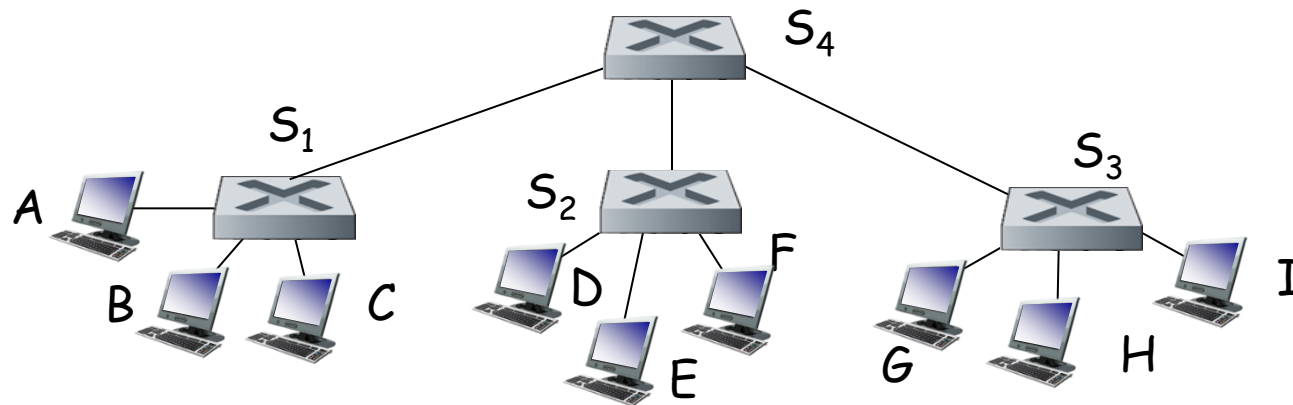


MAC addr	interface	TTL
A	1	60
A'	4	60

Bảng switch
(ban đầu trống)

Kết nối các switch với nhau (Interconnecting switches)

- ❖ Các switch có thể được kết nối với nhau

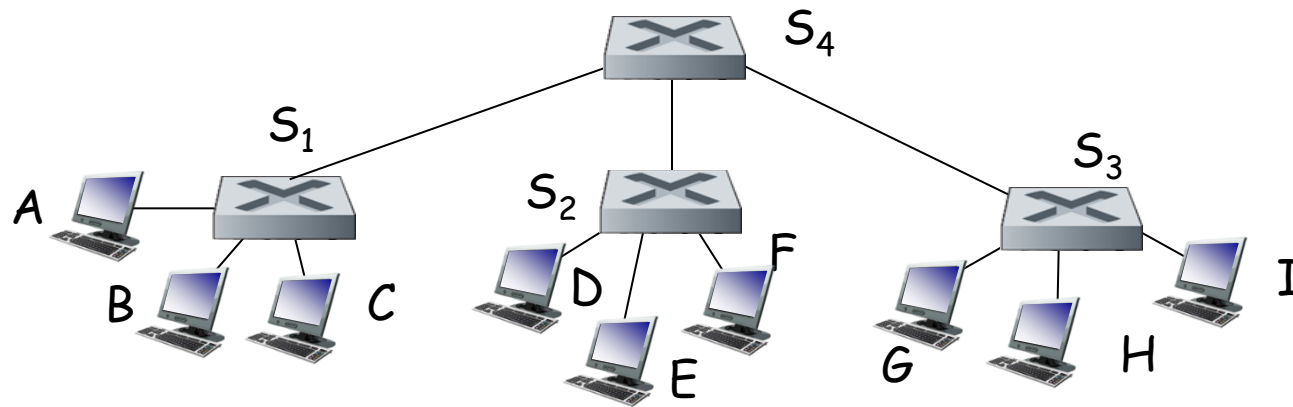


Hỏi: gửi từ A tới G – làm cách nào S₁ biết đề chuyển tiếp frame tới F thông qua S₄ và S₃?

❖ Trả: tự học! (làm việc giống *yl chang* như trong trường hợp chỉ có 1 switch!)

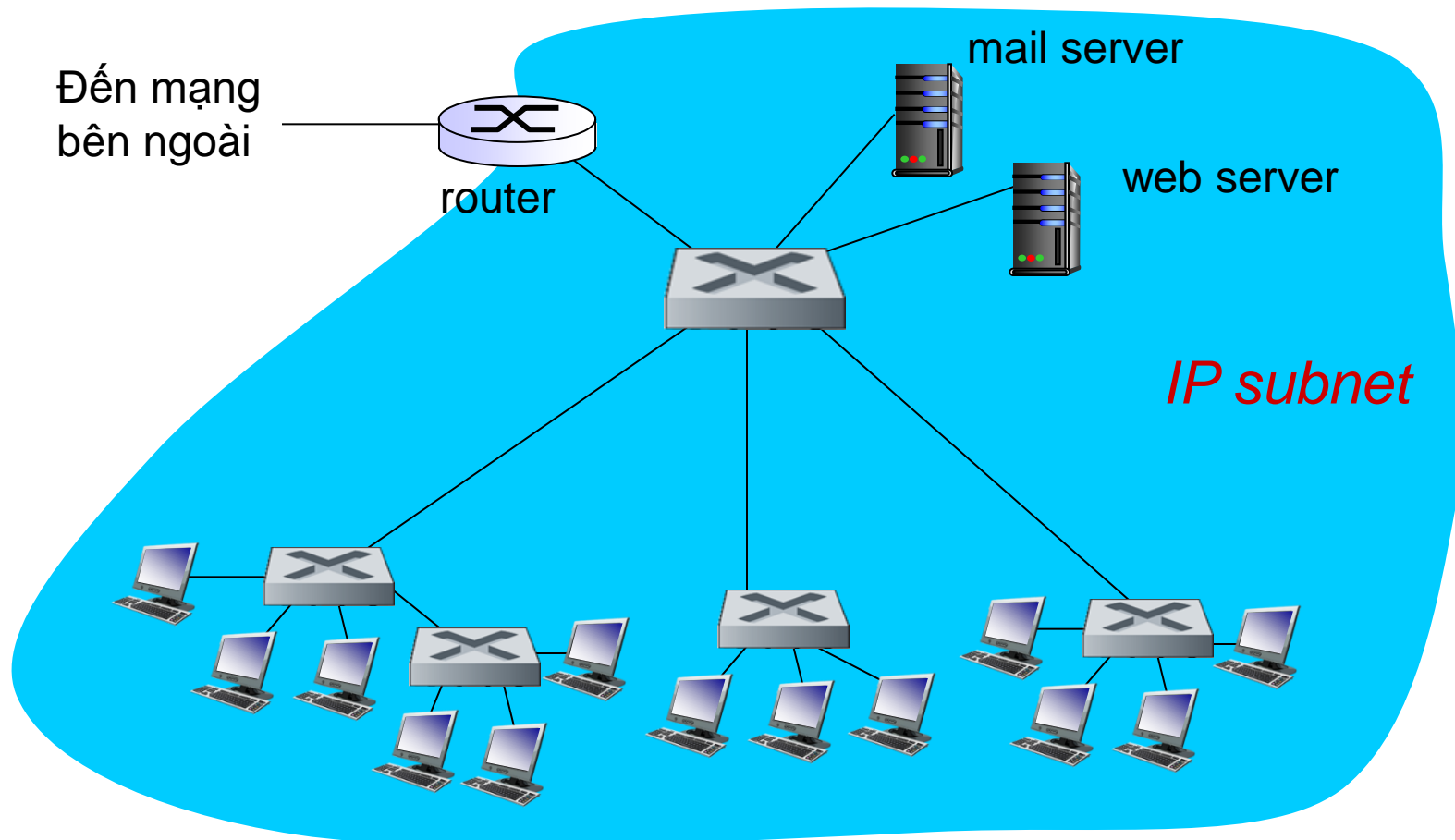
Ví dụ nhiều switch tự học

Giả sử C gửi frame tới I, I trả lời cho C



- ❖ Hỏi: trình bày các bảng của các switch và cách các gói tin được chuyển đi tại các switch S_1 , S_2 , S_3 , S_4

Mạng của tổ chức



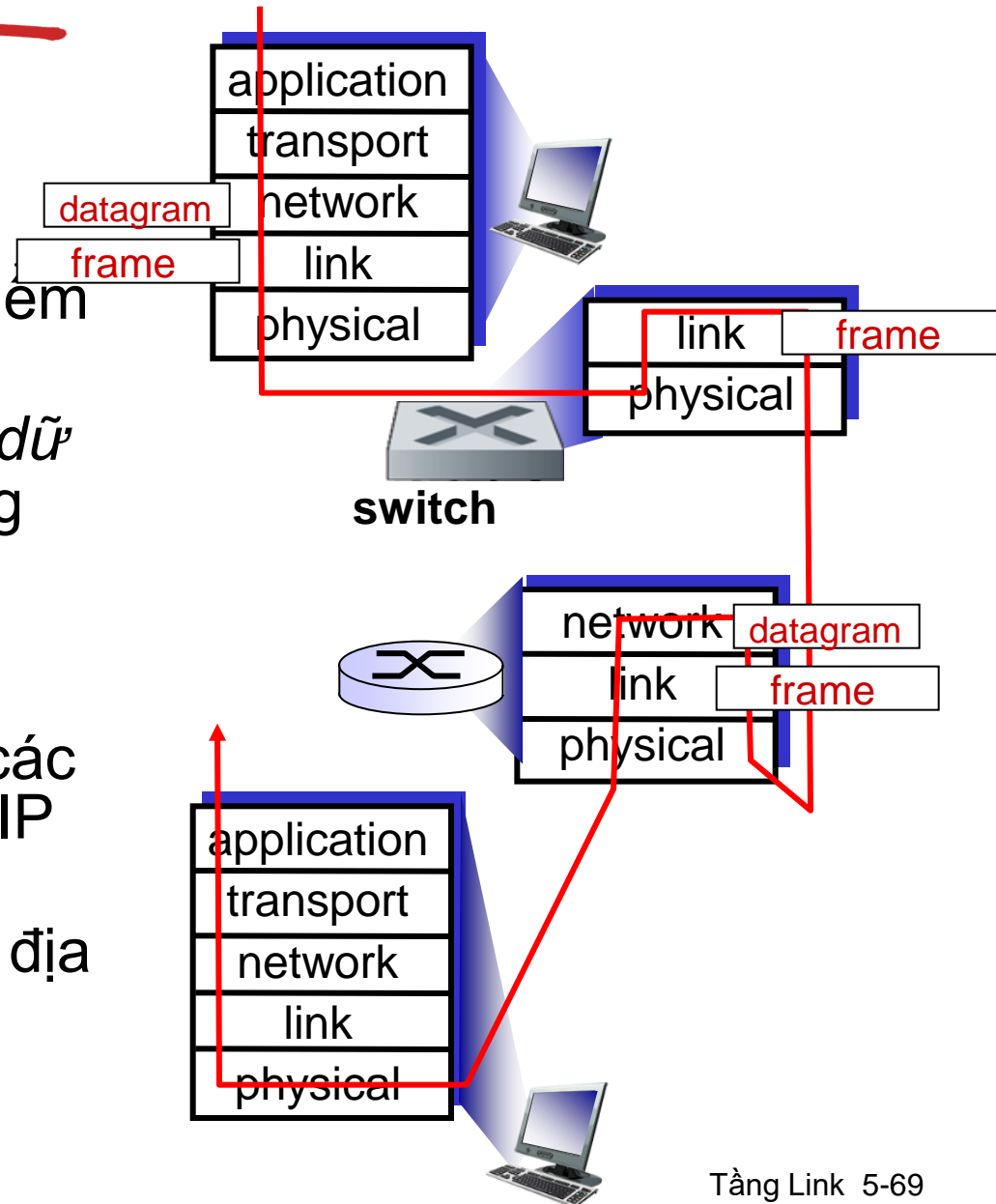
So sánh Switch và Router

Cả 2 đều lưu và chuyển tiếp (store-and-forward):

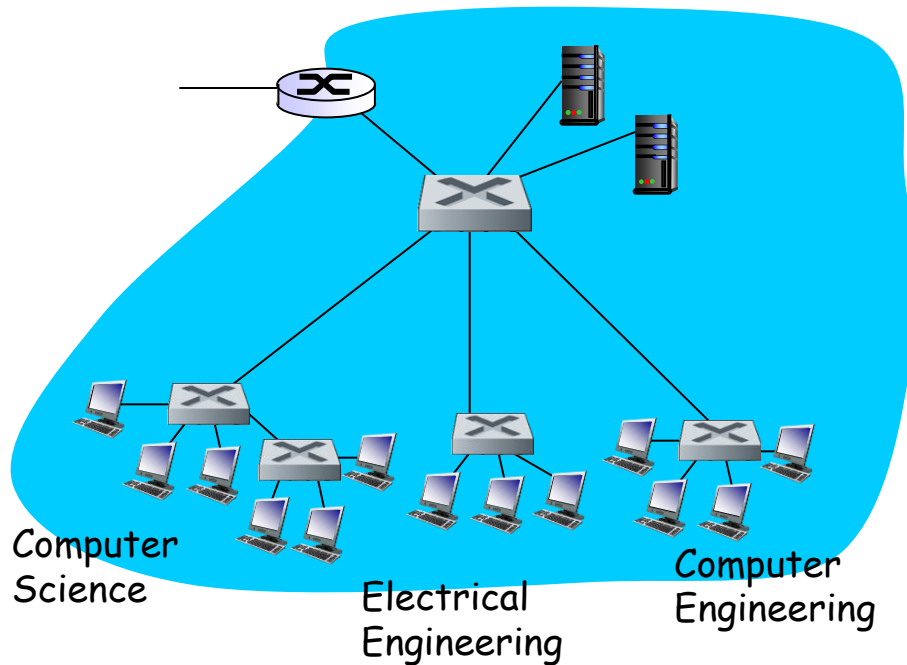
- **router**: thiết bị tầng Mạng (kiểm tra header của tầng Mạng)
- **switch**: thiết bị tầng Liên kết dữ liệu (kiểm tra header của tầng Liên kết dữ liệu)

Cả 2 đều có bảng forwarding:

- **router**: tính toán bảng dùng các thuật toán định tuyến, địa chỉ IP
- **switch**: học bảng forwarding dùng cơ chế flooding, tự học, địa chỉ MAC



VLANs: trình bày



Xem xét:

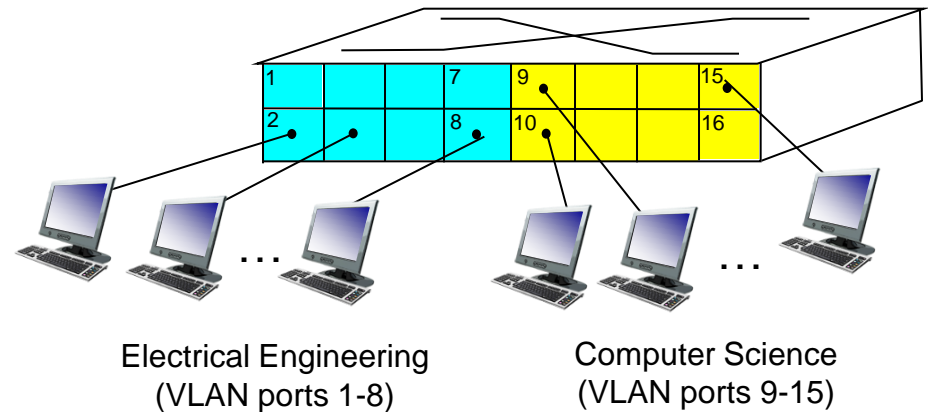
- ❖ Người dùng bên CS di chuyển văn phòng sang EE, nhưng vẫn muốn kết nối CS switch?
- ❖ Miền broadcast đơn:
 - Tất cả lưu lượng broadcast tầng 2 (ARP, DHCP, địa chỉ MAC không biết vị trí đích đến ở đâu) phải đi qua toàn mạng LAN
 - An ninh/riêng tư, các vấn đề về hiệu suất

VLANs

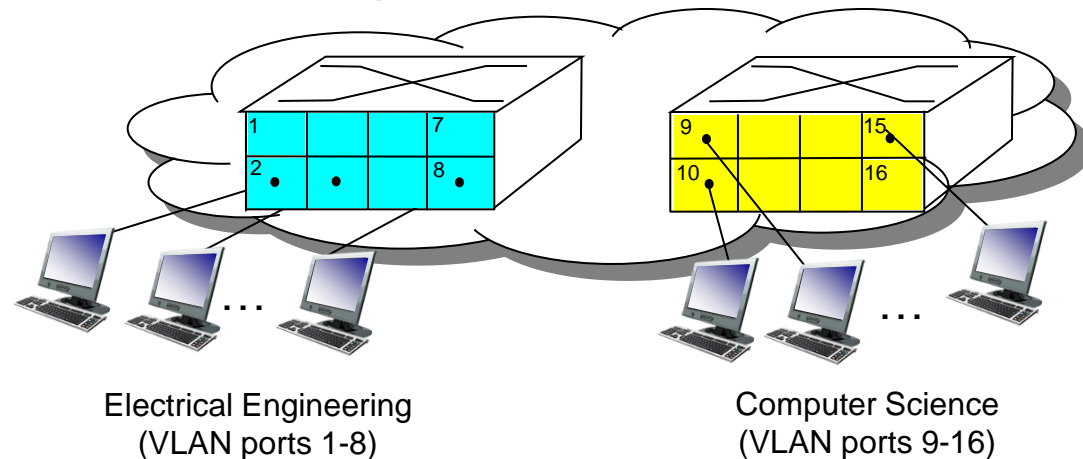
Virtual Local Area Network

Các switch hỗ trợ khả năng VLAN có thể được cấu hình để định nghĩa nhiều mạng LAN ảo (multiple virtual LANS) trên một hạ tầng vật lý của mạng LAN.

port-based VLAN: các port của switch được nhóm lại (bởi phần mềm quản lý switch) để trở thành một switch *vật lý* duy nhất...

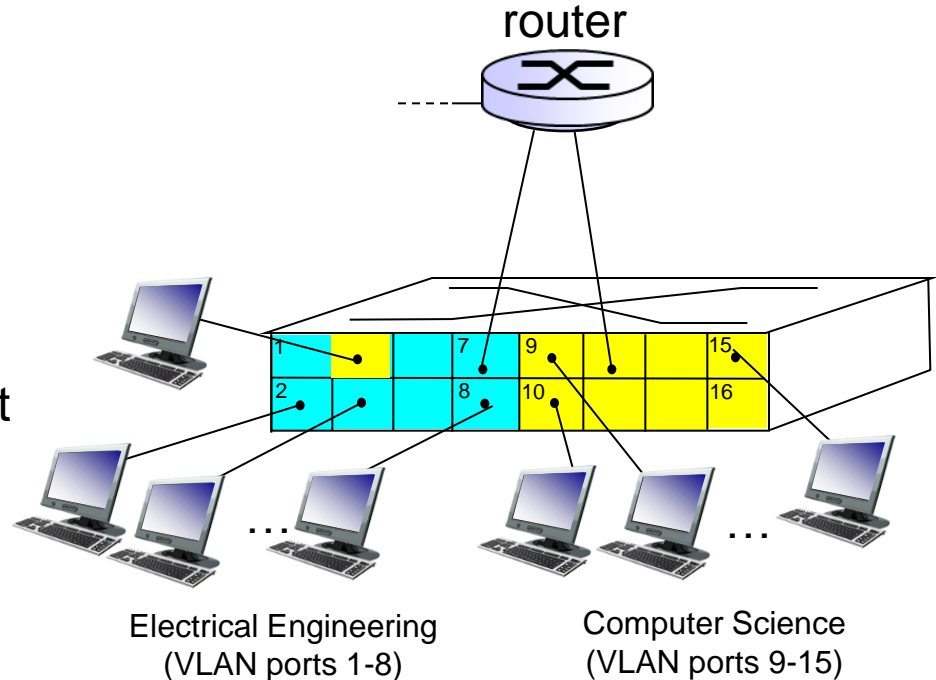


...hoạt động như là *nhiều* switch ảo

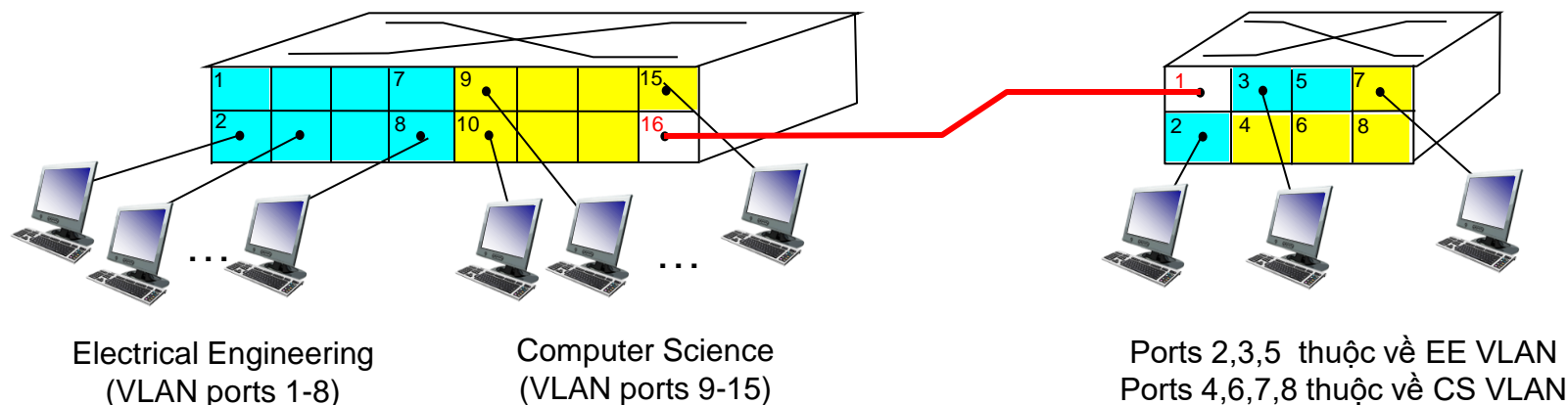


Port-based VLAN

- ❖ *Traffic isolation (cô lập traffic)*: các frame đến từ các port 1-8 chỉ có thể tới được các port 1-8
 - Cũng có thể định nghĩa VLAN dựa trên địa chỉ MAC của thiết bị đầu cuối, hơn là dựa trên port của switch
- ❖ *Thay đổi linh động (dynamic membership)*: các port có thể được gán động giữa các VLAN
- ❖ *Chuyển tiếp giữa các VLAN*: được thực hiện thông qua định tuyến (cũng giống như nối các switch riêng biệt)
 - Trên thực tế, các nhà cung cấp bán các thiết bị switch đã kết hợp thêm một vài tính năng của router

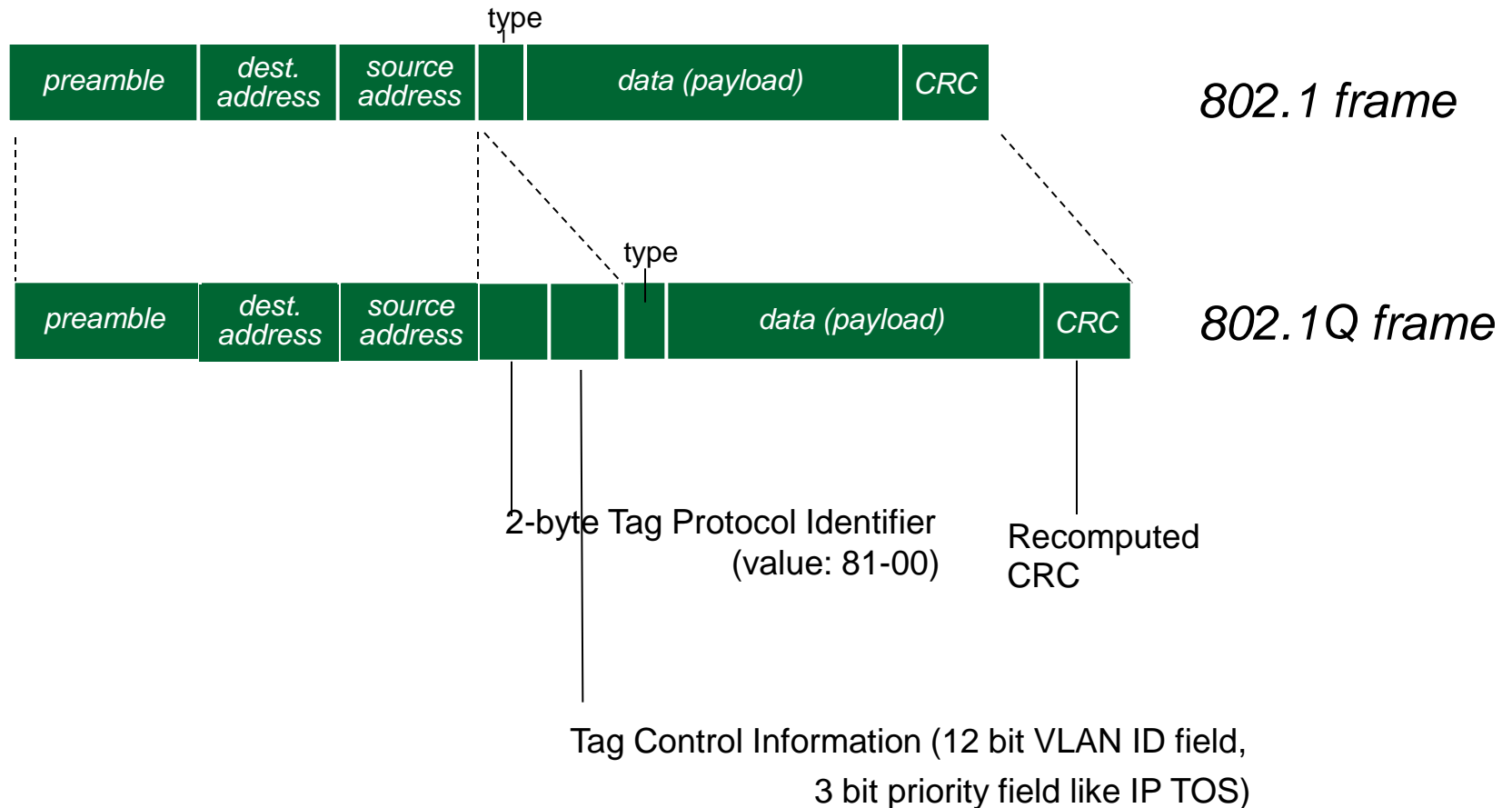


VLANs nối nhiều switch



- ❖ **Trunk port:** mang các frame giữa các VLAN được định nghĩa trên nhiều switch vật lý
 - Các frame được chuyển tiếp bên trong VLAN giữa các switch không thể là các frame 802.1 (phải mang thông tin VLAN ID)
 - Giao thức 802.1q thêm/gỡ bỏ các trường header được thêm vào các frame được chuyển tiếp giữa các trunk port

Định dạng frame VLAN 802.1Q



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

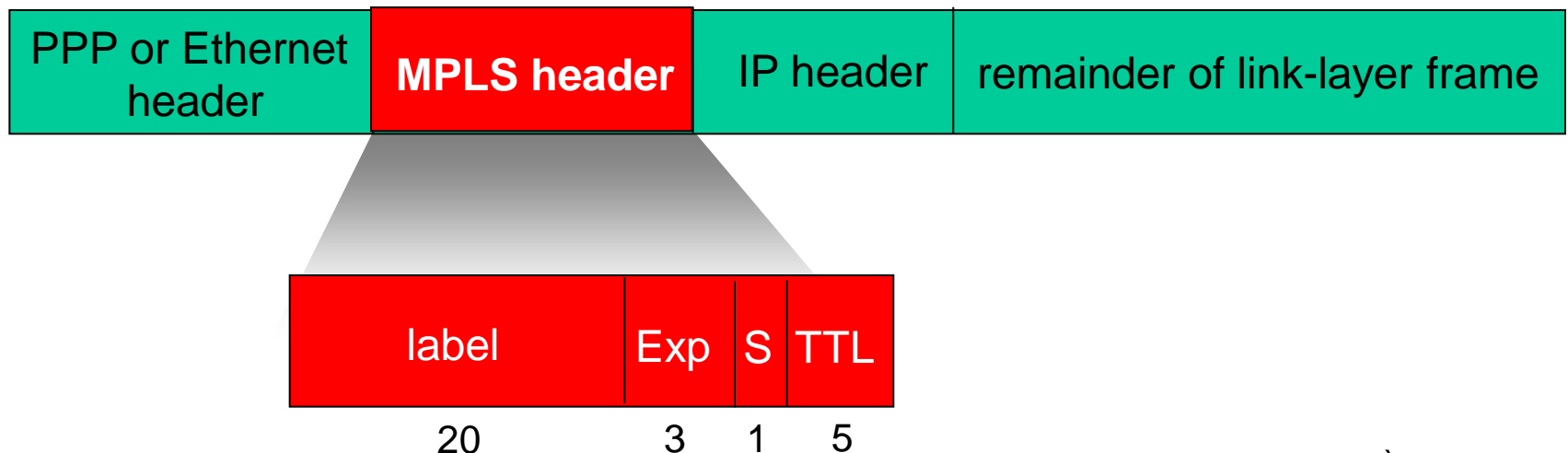
5.5 Link virtualization: MPLS

5.6 Mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Multiprotocol label switching (MPLS)

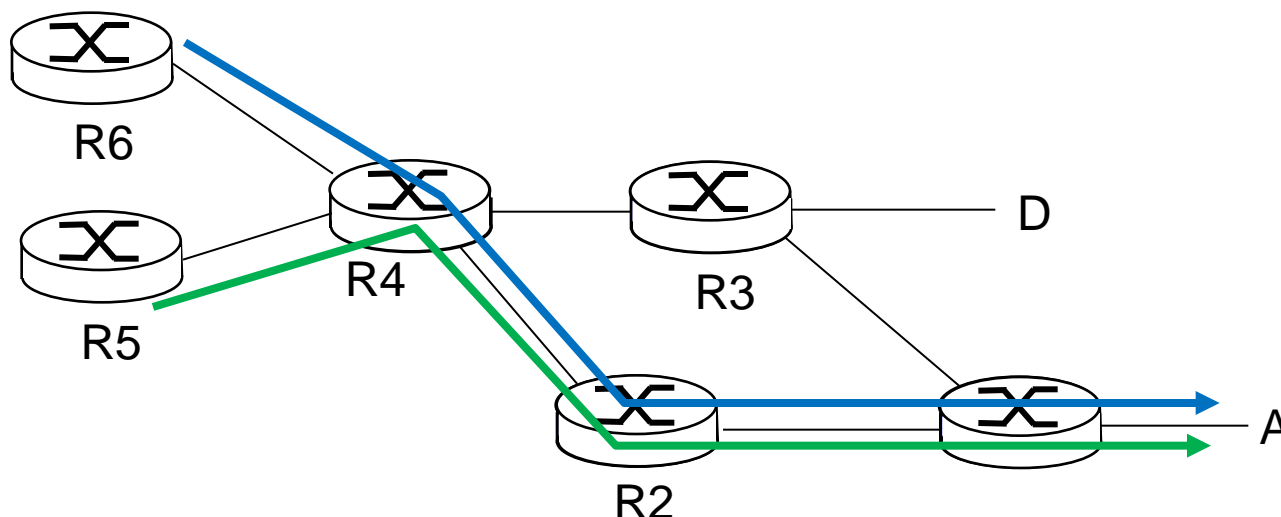
- ❖ Mục tiêu ban đầu: chuyển tiếp gói IP tốc độ cao dùng nhãn có độ dài cố định (fixed length label) (thay cho địa chỉ IP)
 - Tra cứu nhanh dùng định dạng có chiều dài cố định (fixed length identifier) (chứ không so sánh với phần đầu ngắn nhất)
 - Lấy ý tưởng từ hướng tiếp cận của Virtual Circuit (VC)
 - Tuy nhiên IP datagram vẫn giữ địa chỉ IP!



Router có khả năng MPLS

- ❖ Còn gọi là Router chuyển mạch nhãn (label-switched router)
- ❖ Chuyển tiếp các packet tới interface đầu ra chỉ dựa trên giá trị nhãn (label value) (*không kiểm tra địa chỉ IP*)
 - Bảng chuyển tiếp MPLS (MPLS forwarding table) khác với bảng chuyển tiếp IP (IP forwarding tables)
- ❖ **Linh hoạt:** các quyết định chuyển tiếp MPLS có thể khác với IP của chúng
 - Dùng địa chỉ đích và nguồn để định tuyến các luồng dữ liệu tới cùng đích đến một cách khác nhau
 - Định tuyến lại các luồng dữ liệu nhanh chóng nếu đường liên kết hỏng: các đường dẫn dự phòng được tính toán trước (hữu dụng cho VoIP)

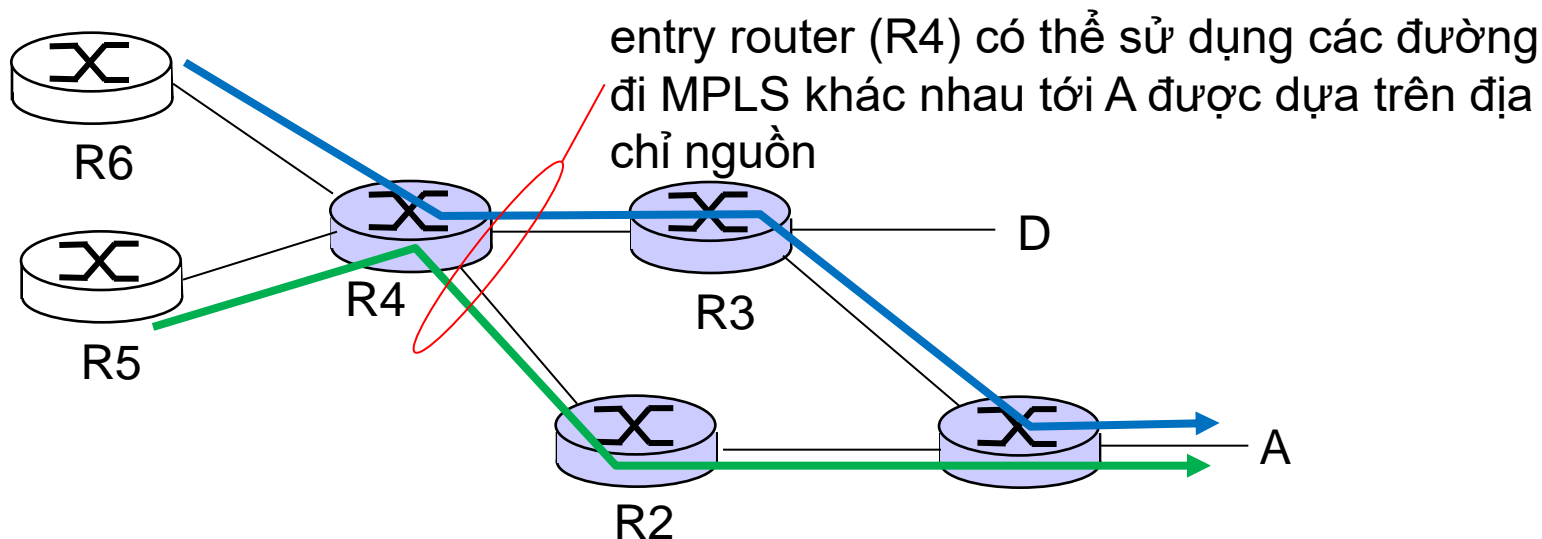
So sánh đường đi MPLS và IP



❖ **Định tuyến IP:** đường tới đích đến được xác định chỉ bởi địa chỉ đích



So sánh đường đi MPLS và IP



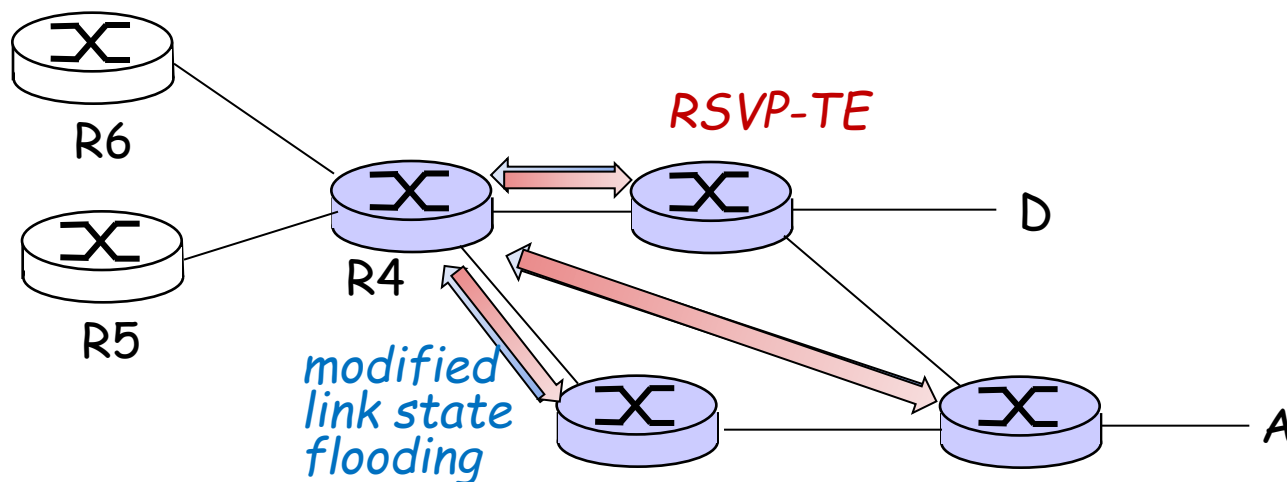
- ❖ **Định tuyến IP:** đường tới đích đến được xác định chỉ bởi địa chỉ đích
- ❖ **Định tuyến MPLS:** đường tới đích đến có thể được dựa trên địa chỉ nguồn và đích
 - **Định tuyến lại nhanh chóng (fast reroute):** tính toán lại các đường đi dự phòng trong trường hợp đường kết nối bị hỏng

 IP-only router

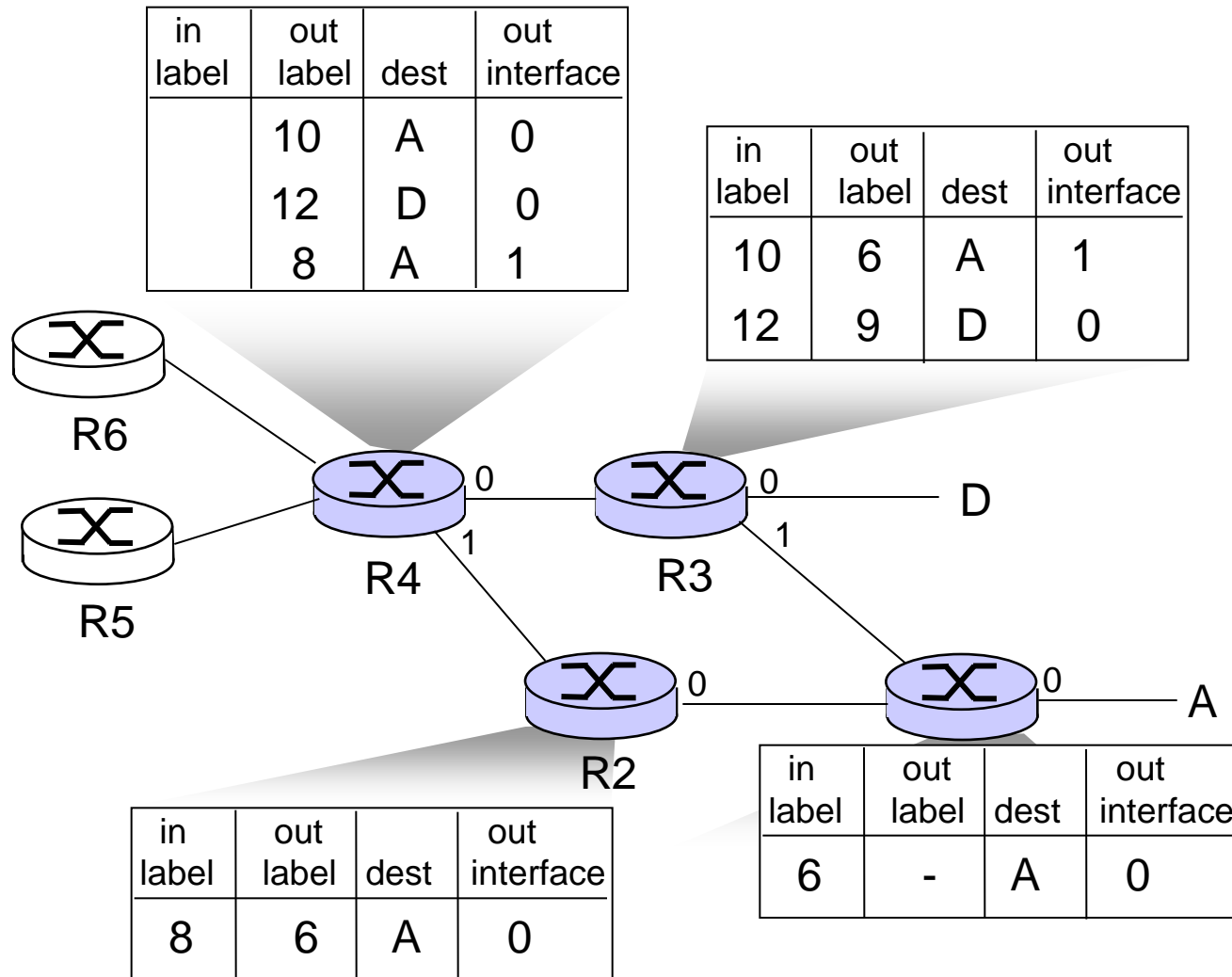
 MPLS và IP router

Tín hiệu MPLS

- ❖ Chỉnh sửa các giao thức flooding IS-IS link-state, OSPF để mang thông tin được sử dụng bởi định tuyến MPLS,
 - Ví dụ: link bandwidth, số lượng băng thông đường link “được dành riêng”
- ❖ *entry của router MPLS sử dụng giao thức tín hiệu RSVP-TE để thiết lập chuyển tiếp MPLS tại các router nhận*



Bảng chuyển tiếp MPLS



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

5.5 Link virtualization: MPLS

5.6 Mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

Mạng trung tâm dữ liệu

- ❖ Hàng chục đến hàng trăm ngàn host:
 - e-business (e.g. Amazon)
 - content-servers (như là YouTube, Akamai, Apple, Microsoft)
 - search engines, data mining (như là Google)
- ❖ Thách thức:
 - Nhiều ứng dụng, mỗi cái phục vụ số lượng lớn client
 - Quản lý/cân bằng tải, tránh tắc nghẽn trong việc xử lý dữ liệu, mạng

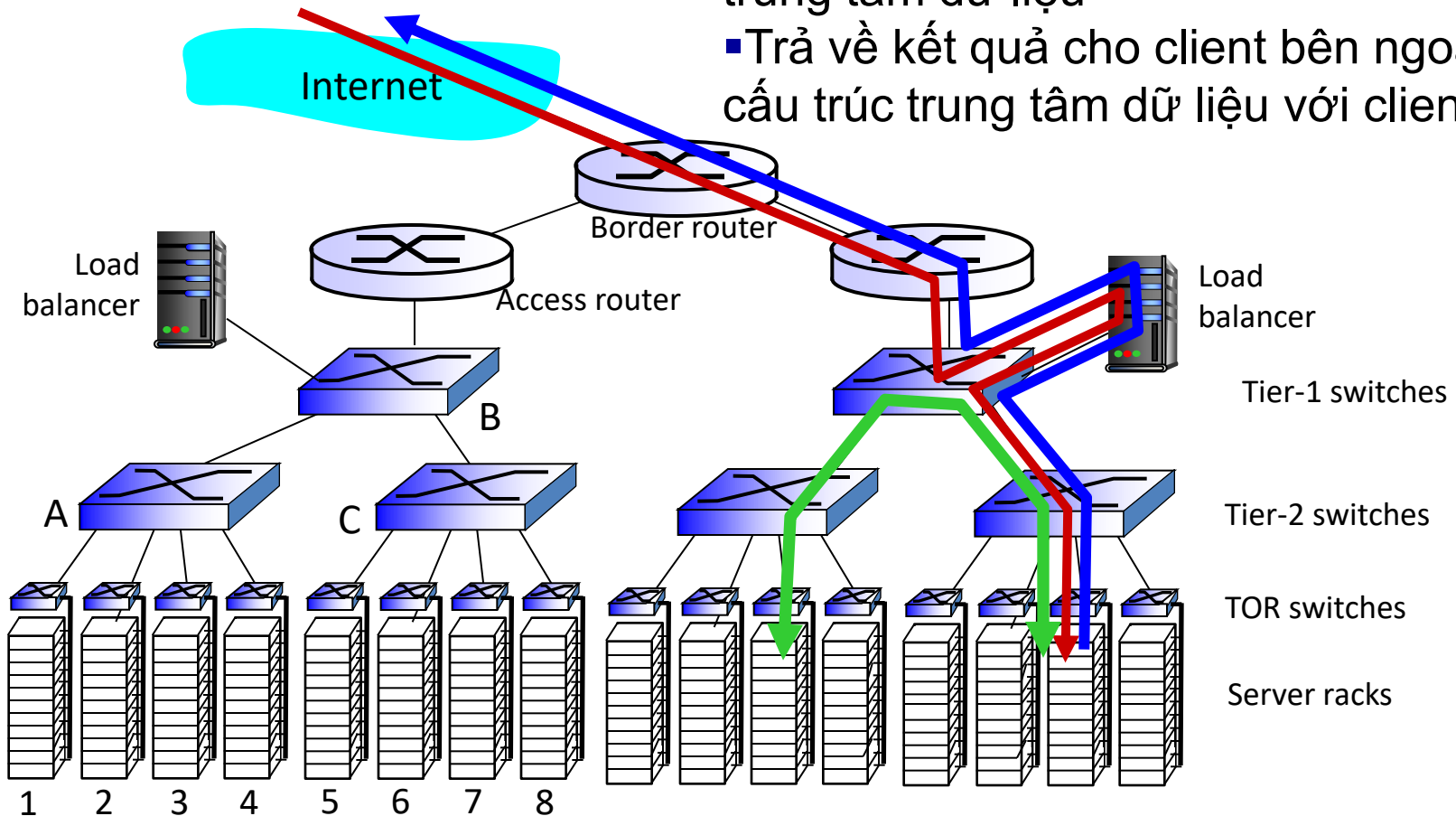


Inside a 40-ft Microsoft container,
Chicago data center

Mạng trung tâm dữ liệu

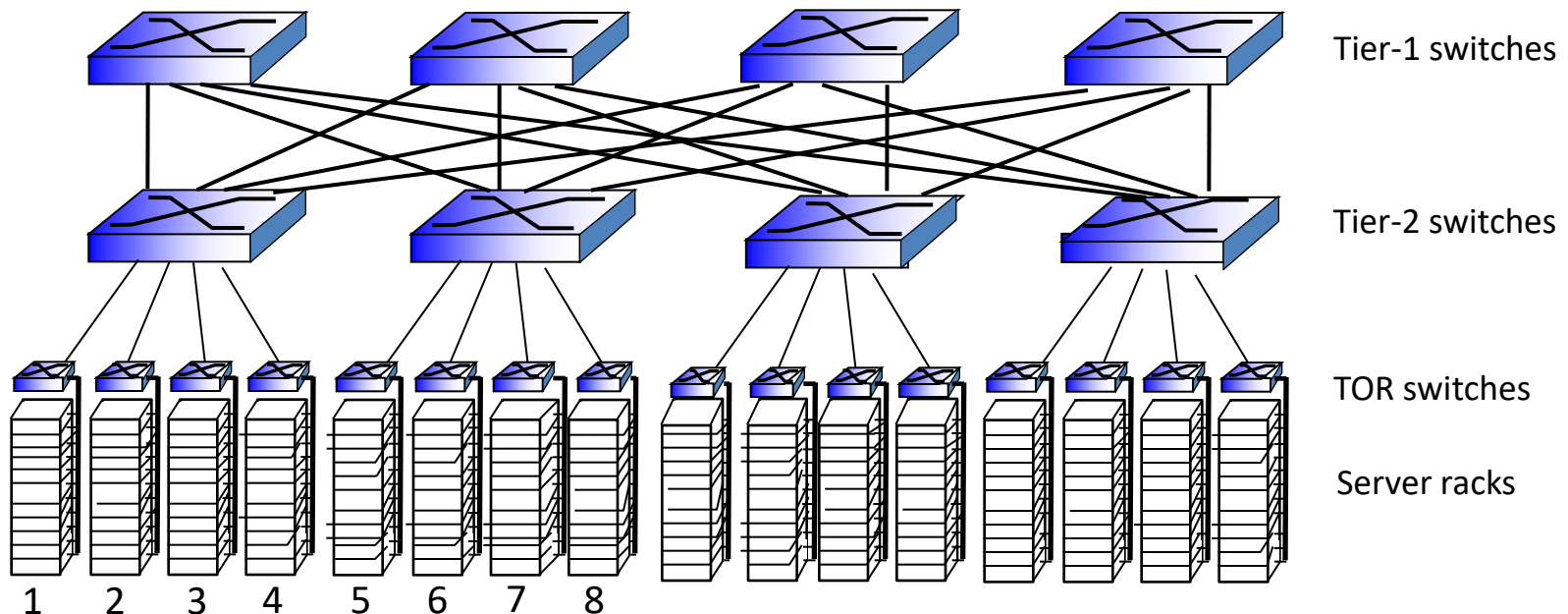
Cân bằng tải: định tuyến tầng application

- Nhận các yêu cầu từ client bên ngoài
- Điều phối khối lượng công việc trong trung tâm dữ liệu
- Trả về kết quả cho client bên ngoài (che cấu trúc trung tâm dữ liệu với client)



Mạng trung tâm dữ liệu

- ❖ Rất nhiều kết nối giữa các switch và rack:
 - Thông lượng được tăng lên giữa các rack (nhiều đường định tuyến có thể dùng được)
 - Độ tin cậy và khả năng dự phòng tăng lên



Tầng Liên kết dữ liệu và mạng LAN

5.1 Giới thiệu và các dịch vụ

5.2 Phát hiện lỗi và sửa lỗi

5.3 Các giao thức đa truy cập

5.4 Mạng LAN

- Định địa chỉ, ARP
- Ethernet
- Switches
- VLANs

5.5 Link virtualization: MPLS

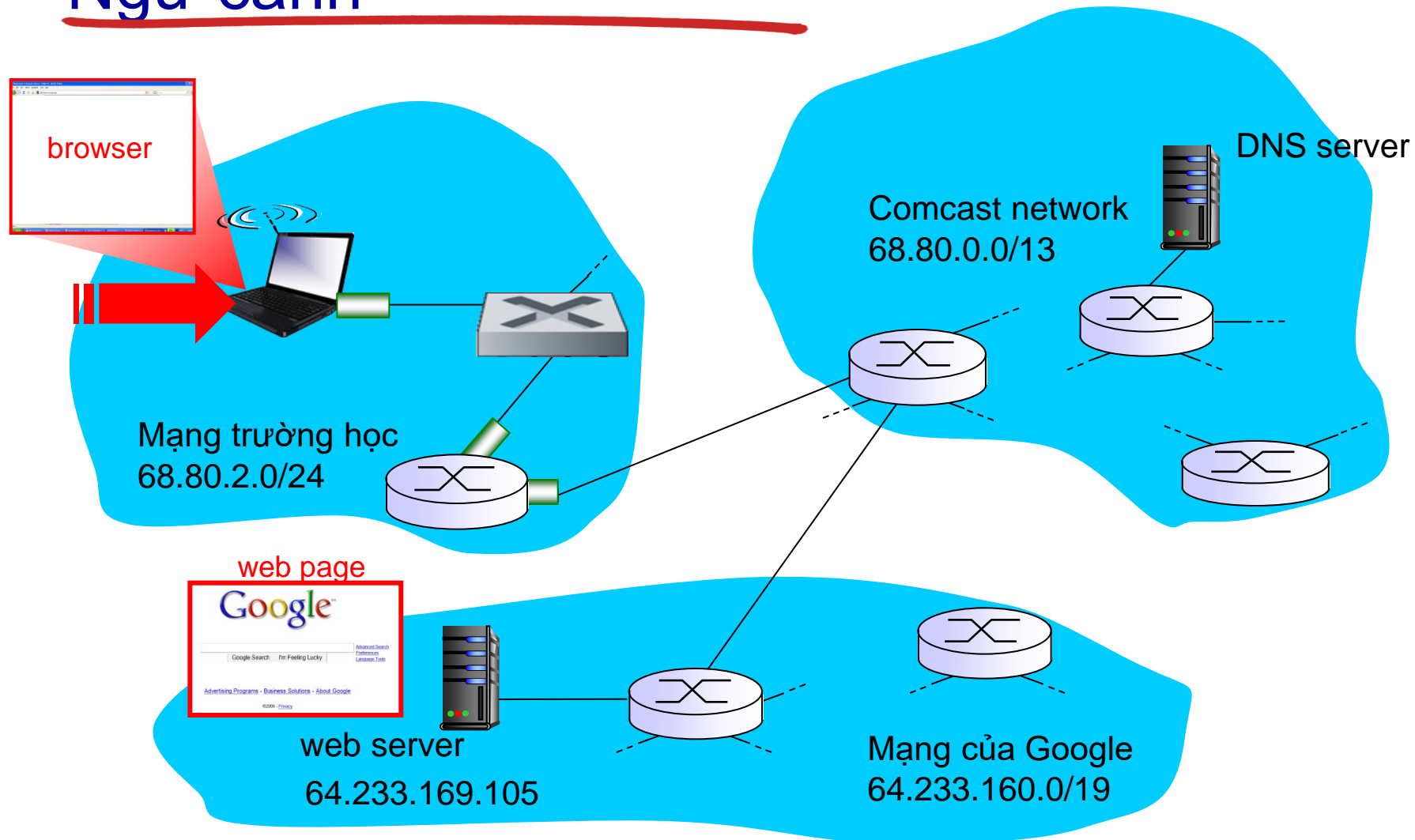
5.6 Mạng trung tâm dữ liệu

5.7 Hoạt động thường ngày - truy cập web

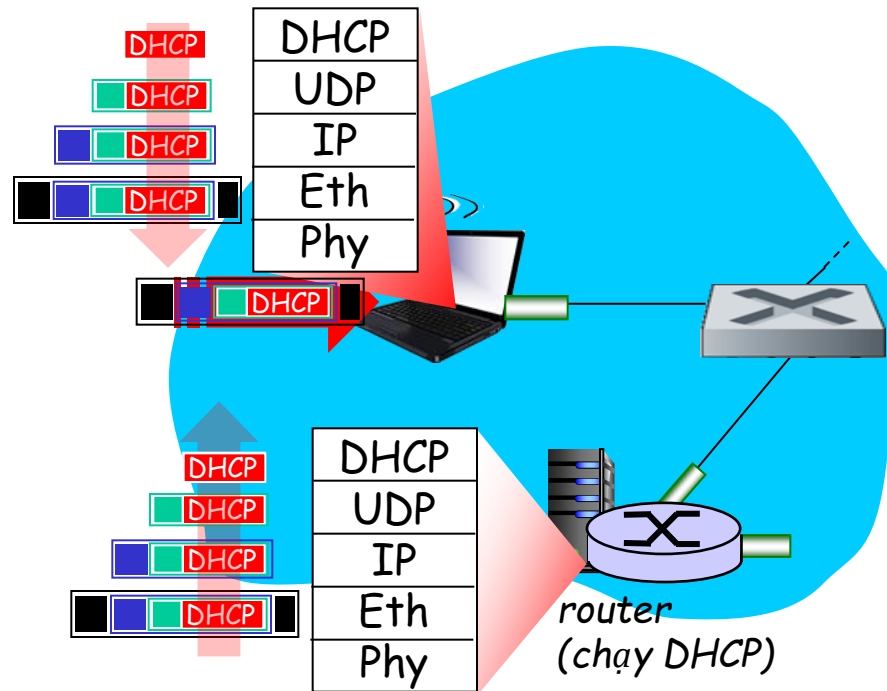
Tổng hợp: một truy vấn web thường ngày

- ❖ Tìm hiểu *đầy đủ* chồng giao thức từ trên xuống dưới!
 - application, transport, network, link
- ❖ Tổng hợp toàn bộ quá trình
 - *Mục tiêu:* xác định, xem xét, hiểu các giao thức (tại tất cả các tầng) được tham gia vào tình huống đơn giản: truy vấn trang www
 - *Ngữ cảnh:* sinh viên kết nối máy tính xách tay vào mạng của tòa nhà trường học, gửi yêu cầu/nhận phản hồi tới/từ www.google.com

Ngữ cảnh

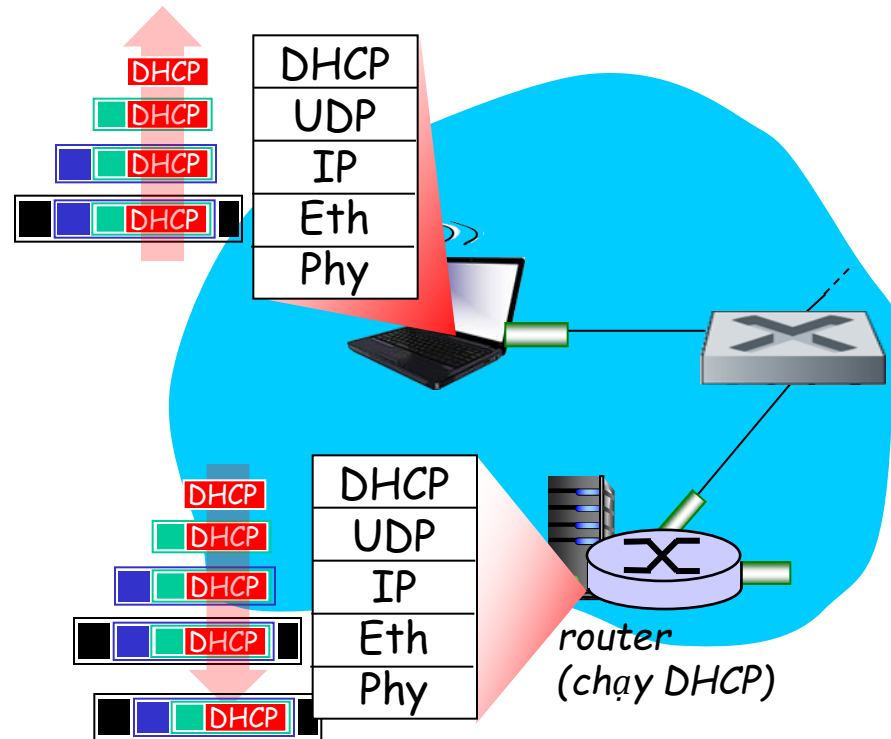


Thường nhật ... kết nối tới Internet



- ❖ Máy tính kết nối cần có địa chỉ IP của riêng nó, địa chỉ của cổng ra mặc định, địa chỉ của DNS server: dùng **DHCP**
- ❖ Yêu cầu DHCP được đóng gói trong **UDP**, được đóng gói trong **IP**, được đóng gói trong **802.3** Ethernet
- ❖ Ethernet frame **broadcast** (dest: FFFFFFFFFFFFFFFF) trên LAN, được router có chạy dịch vụ **DHCP** nhận
- ❖ Ethernet **mở gói**, chuyển tiếp gói IP lên, IP mở gói, chuyển tiếp gói UDP lên, UDP mở gói, chuyển tiếp gói yêu cầu DHCP lên DHCP

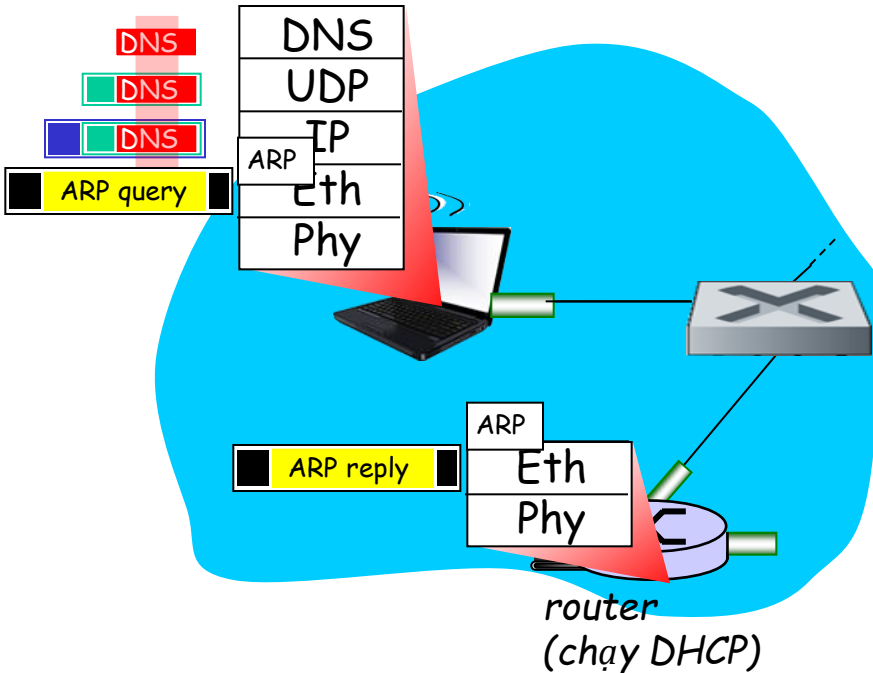
Thường nhật ... kết nối tới Internet



- ❖ DHCP server tạo **DHCP ACK** chứa địa chỉ IP của client, địa chỉ IP của cổng ra mặc định cho client đó, địa chỉ IP của DNS server
- ❖ Frame được đóng gói tại DHCP server, và được chuyển tiếp (**switch tự học**) thông qua mạng LAN, rồi tách ra tại client
- ❖ DHCP client nhận gói trả lời DHCP ACK

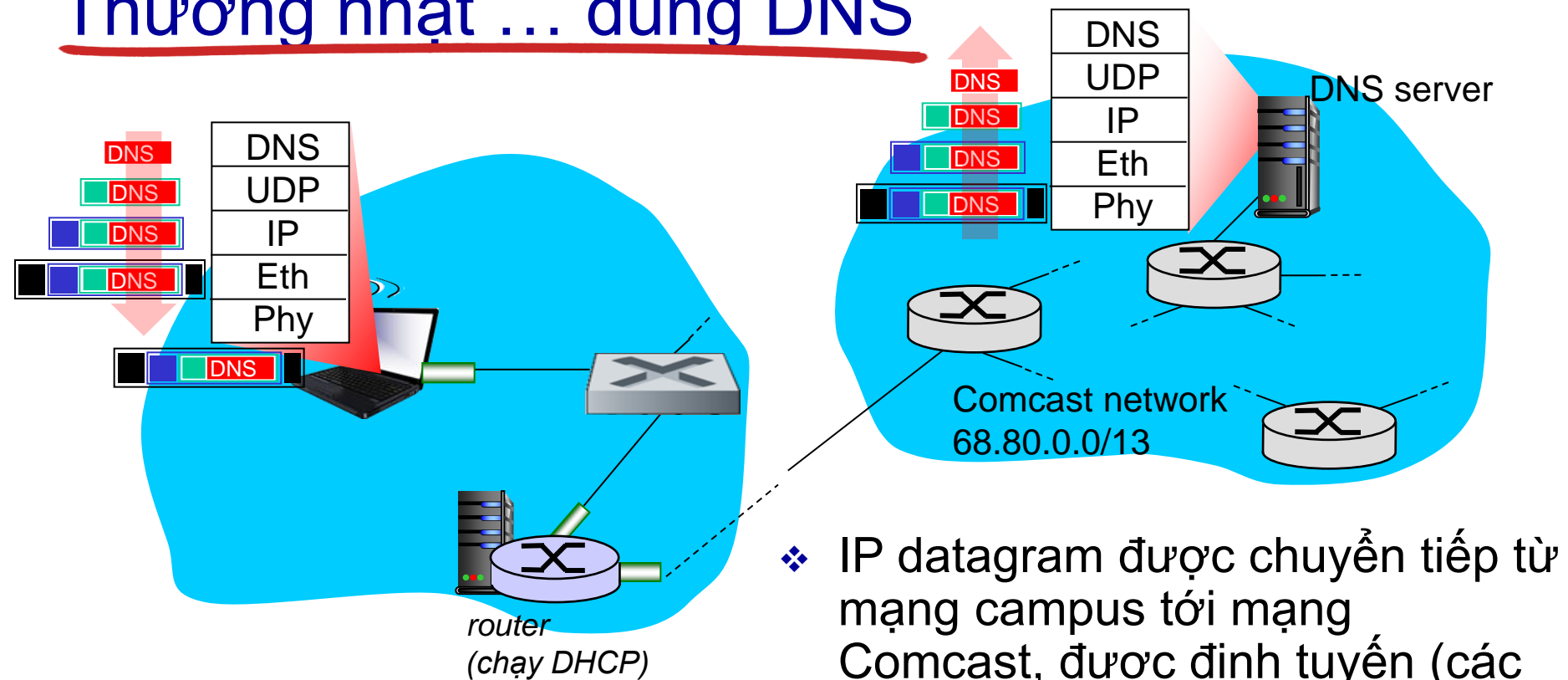
Bây giờ, Client có địa chỉ, biết địa chỉ của DNS server, địa chỉ IP của router cổng ra mặc định

Thường nhật ... ARP (trước DNS, trước HTTP)



- ❖ Trước khi gửi *HTTP* request, cần địa chỉ IP của `www.google.com`: *DNS*
- ❖ DNS query được tạo, đóng gói trong UDP, được đóng gói trong IP, được đóng gói trong Eth. Gửi frame tới router, cần địa chỉ MAC của interface của router interface: *ARP*
- ❖ *ARP query* broadcast, được nhận bởi router, router này sẽ trả lời lại với *ARP reply* cung cấp địa chỉ MAC của interface của router này
- ❖ Hiện tại, client biết địa chỉ MAC của router cổng ra mặc định, vì vậy nó có thể gửi frame chứa DNS query

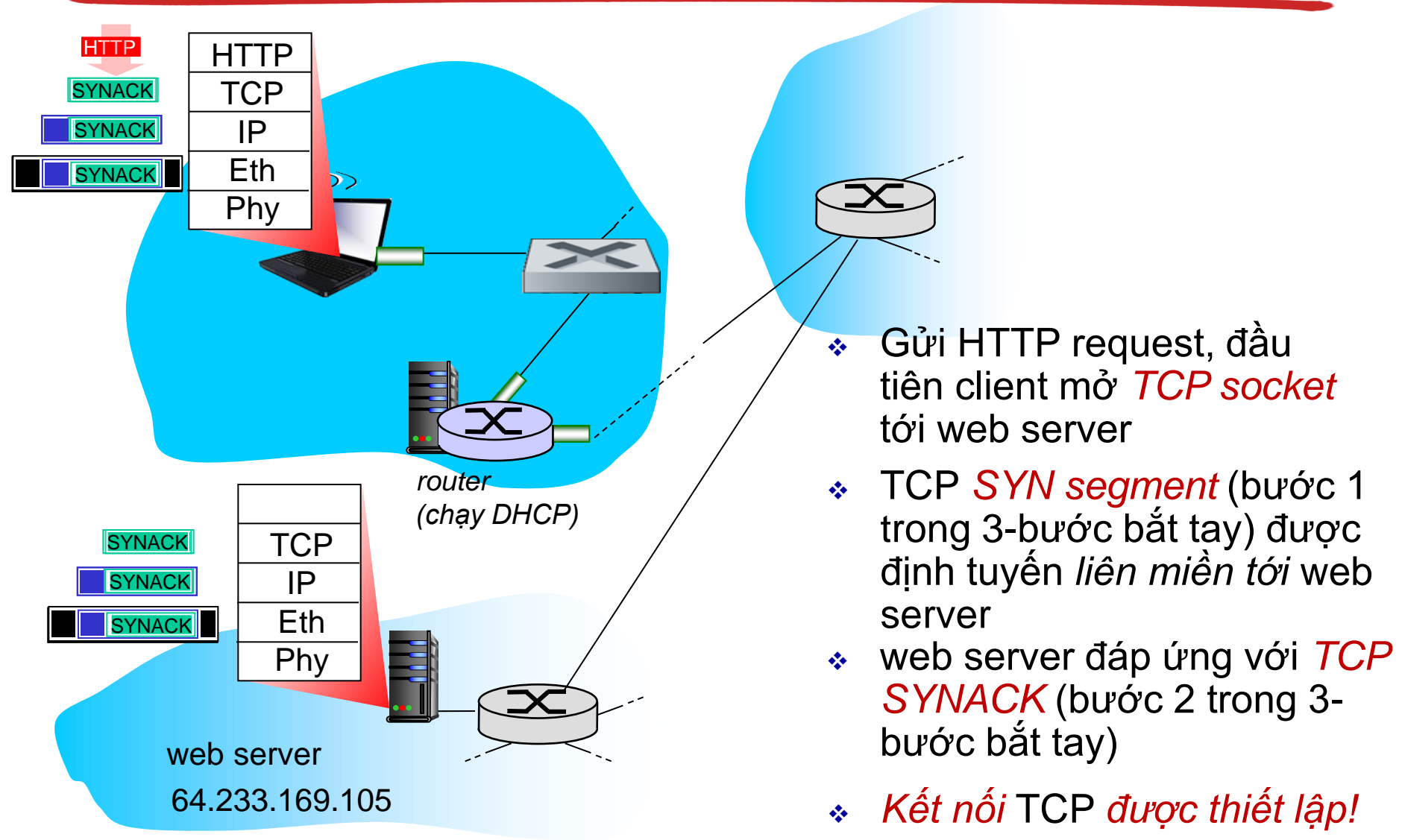
Thường nhật ... dùng DNS



- ❖ IP datagram chứa DNS query được chuyển tiếp thông qua switch của mạng LAN từ client tới router cổng ra mặc định

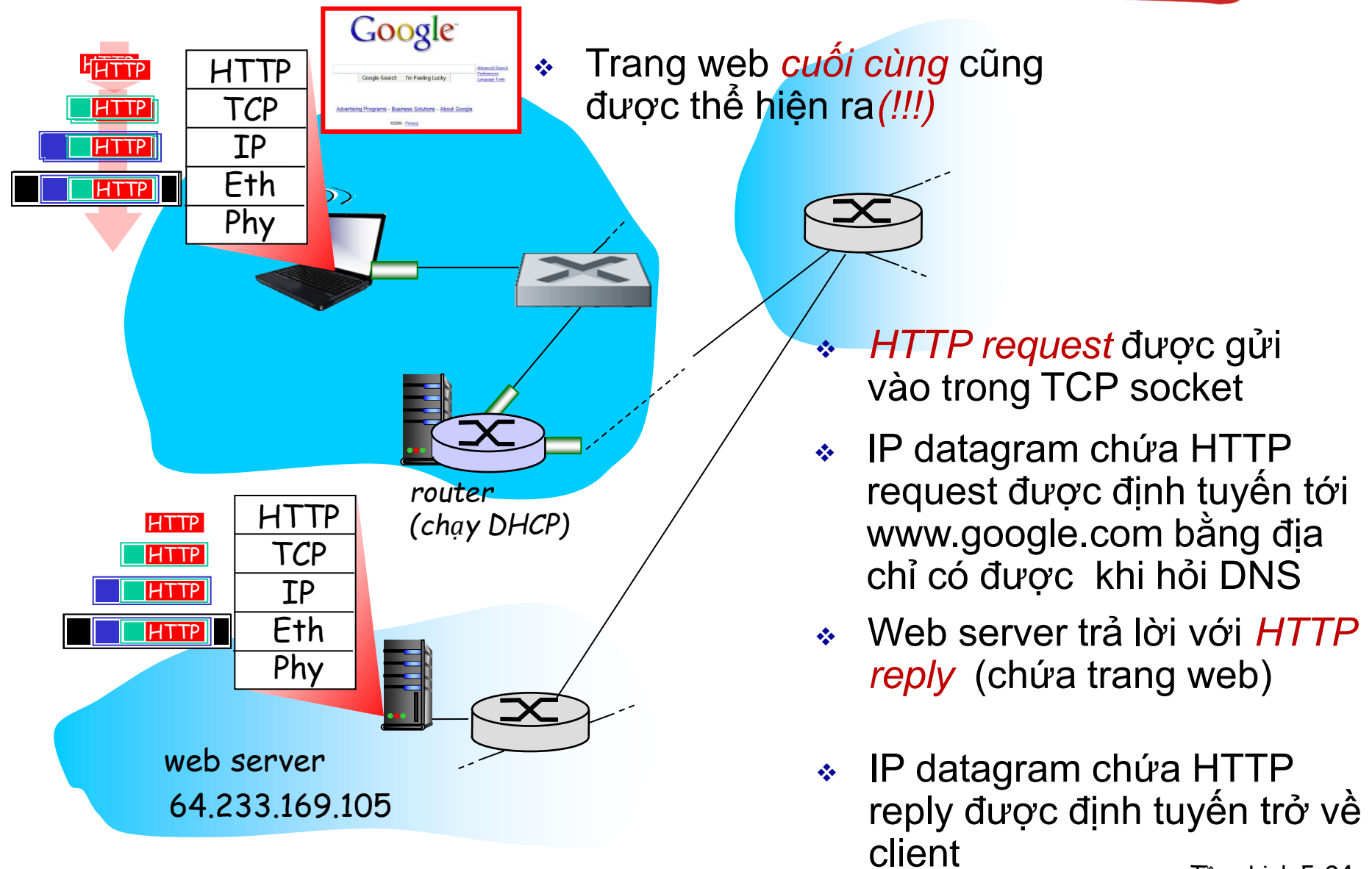
- ❖ IP datagram được chuyển tiếp từ mạng campus tới mạng Comcast, được định tuyến (các bảng được tạo bởi các giao thức định tuyến **RIP, OSPF, IS-IS** và/hoặc **BGP**) tới DNS server
- ❖ Được tách/ghép tới DNS server
- ❖ DNS server trả lời cho client với địa chỉ IP address của **www.google.com**

Thường nhật ... kết nối TCP mang HTTP



Thường nhật ...

HTTP yêu cầu/trả lời (request/reply)



Chương 5: Tổng kết

- ❖ Các nguyên lý của các dịch vụ tầng Liên kết dữ liệu:
 - Phát hiện và sửa chữa lỗi
 - Chia sẻ kênh broadcast: đa truy cập
 - Định địa chỉ tầng Liên kết dữ liệu
- ❖ Triển khai các công nghệ khác nhau của tầng Liên kết dữ liệu
 - Ethernet
 - Mạng LAN và VLAN chuyển mạch
 - Mạng ảo hóa như là một tầng thuộc tầng Liên kết dữ liệu: MPLS
- ❖ Tổng hợp: một truy vấn web thường nhật.

Chương 5: kết thúc

- ❖ Tìm hiểu *đầy đủ* chồng giao thức từ trên xuống dưới (ngoại trừ PHY)
- ❖ Hiểu về các nguyên lý hoạt động của mạng và hiện thực
- ❖ Có thể dừng tại đây Nhưng có một số chủ đề thú vị!
 - wireless
 - multimedia
 - security
 - network management