

# Mạng máy tính

**TS. Phạm Tuấn Minh**

Khoa Công nghệ Thông tin, Đại học Thủy lợi

[minhpt@tlu.edu.vn](mailto:minhpt@tlu.edu.vn)

<http://netlab.tlu.edu.vn/~minhpt/>

# Chương 5: Tầng liên kết

## 5.1 Giới thiệu

## 5.2 Phát hiện lỗi, sửa lỗi

## 5.3 Các giao thức đa truy cập

## 5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

## 5.5 Ảo hóa liên kết: MPLS

## 5.6 Mạng trung tâm dữ liệu

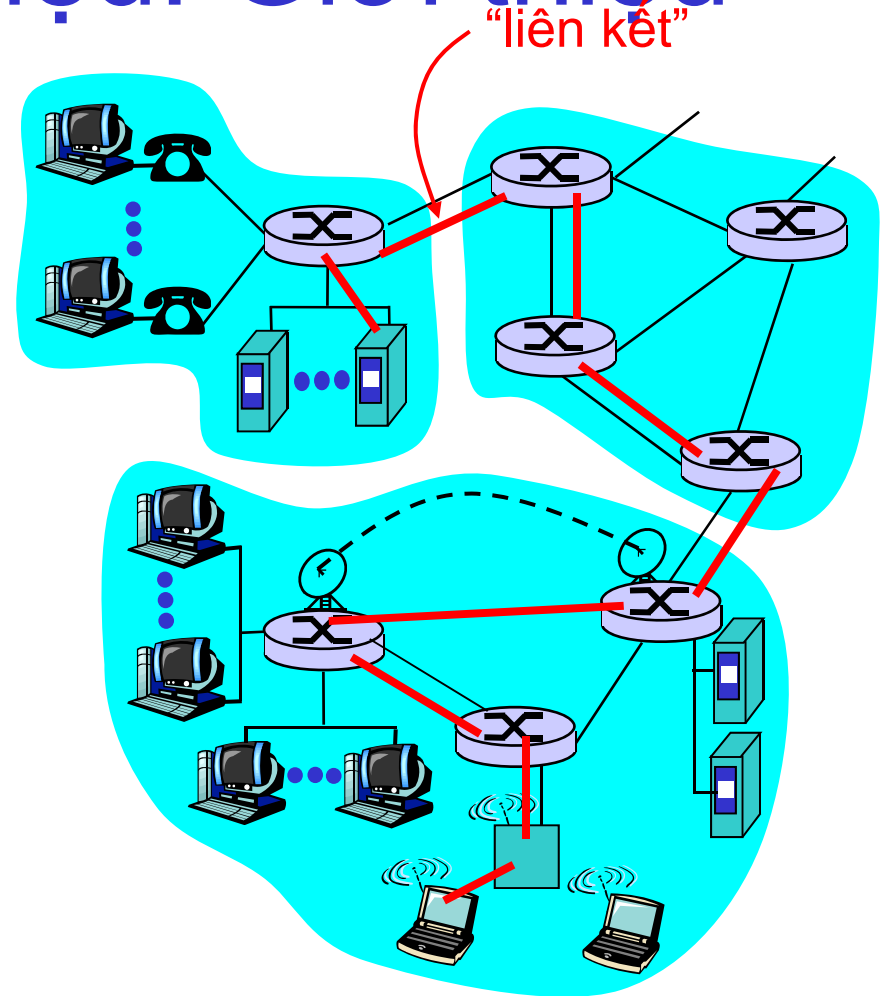
# Tầng liên kết dữ liệu

- ❑ 5.1 Giới thiệu và các dịch vụ
- ❑ 5.2 Phát hiện và sửa lỗi
- ❑ 5.3 Các giao thức đa truy cập
- ❑ 5.4 Đánh địa chỉ tầng liên kết dữ liệu
- ❑ 5.5 Ethernet
- ❑ 5.6 Hub và switch
- ❑ 5.7 PPP
- ❑ 5.8 Link Virtualization: ATM và MPLS

# Tầng liên kết dữ liệu: Giới thiệu

## Một số thuật ngữ:

- host và router: **nút**
- Kênh truyền thông kết nối các nút kề dọc theo đường truyền thông gọi là **liên kết (link)**
  - Liên kết có dây
  - Liên kết không dây
  - LAN
- Các gói tin tầng 2 gọi là **khung (frame)**, đóng gói các datagram



**Tầng liên kết dữ liệu** có vai trò truyền các datagram từ một nút tới nút kề qua một liên kết

# Tầng liên kết dữ liệu: Xem xét với tầng khác

- ❑ Datagram được truyền bởi các giao thức liên kết dữ liệu khác nhau qua các liên kết khác nhau:
  - Ví dụ: Ethernet ở liên kết đầu tiên, frame relay ở liên kết giữa, 802.11 ở liên kết cuối cùng
- ❑ Mỗi giao thức liên kết dữ liệu cung cấp các dịch vụ khác nhau
  - Ví dụ: có thể hoặc không cung cấp truyền tin cậy qua liên kết

## Tương tự vận tải:

- ❑ Chuyển đi từ Hà Nội tới Cần Thơ
  - Đường sắt (tàu hỏa): HN -> Đà Nẵng
  - Đường không (máy bay): Đà Nẵng -> Tp HCM
  - Đường bộ (ô tô): Tp HCM -> Cần Thơ
- ❑ khách du lịch = **datagram**
- ❑ transport segment = **liên kết truyền thông (communication link)**
- ❑ hình thức vận tải = **giao thức tầng liên kết dữ liệu**
- ❑ đại lý du lịch = **thuật toán dẫn đường**

# Các dịch vụ tầng liên kết dữ liệu

## ❑ Đóng khung, truy cập liên kết:

- Đóng gói datagram thành các khung, thêm header, trailer
- Truy cập kênh nếu phương tiện dùng chung (shared medium)
- Các địa chỉ “MAC” sử dụng trong header của khung để định danh nguồn, đích
  - Khác với địa chỉ IP!

## ❑ Truyền tin cậy giữa các nút kề

- Đã học cách để thực hiện truyền tin cậy (chương 3)!
- Ít khi sử dụng trên các liên kết lỗi bit thấp (ví dụ: cáp quang, cáp xoắn đôi)
- Liên kết không dây: tỷ lệ lỗi cao
  - Q: Tại sao xét tin cậy cả mức liên kết dữ liệu và end-end?

# Các dịch vụ tầng liên kết dữ liệu

## ❑ *Điều khiển luồng:*

- Tốc độ giữa nút nhận và nút gửi kề nhau

## ❑ *Phát hiện lỗi:*

- Các lỗi gây ra bởi suy hao và nhiễu tín hiệu
- Bên nhận phát hiện sự tồn tại của lỗi:
  - Báo hiệu cho bên gửi biết để gửi lại hoặc loại bỏ khung

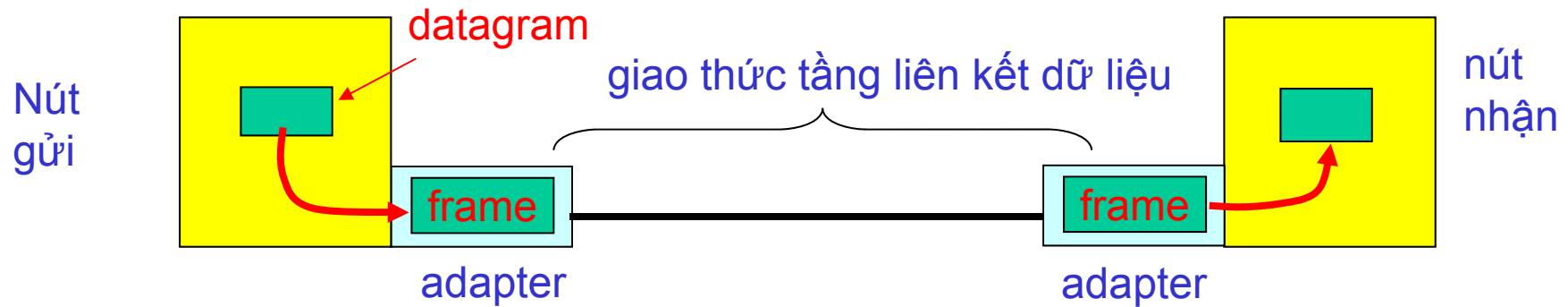
## ❑ *Sửa lỗi:*

- Bên nhận xác định và *sửa* các bit lỗi không phải sử dụng đến việc truyền lại

## ❑ *Half-duplex và full-duplex*

- Với half duplex, các nút tại cả hai điểm cuối của liên kết có thể truyền nhưng không tại cùng thời điểm

# Truyền dữ liệu giữa adaptor



- ❑ Tầng liên kết dữ liệu được cài đặt trong “adaptor” (còn gọi là NIC)
  - Ethernet card, PCMCIA card, 802.11 card

- ❑ Bên gửi:
  - Đóng gói datagram trong một frame
  - Thêm các bit kiểm tra lỗi, rdt, điều khiển luồng,...

- ❑ Bên nhận
  - Tìm kiếm lỗi, rdt, điều khiển luồng,...
  - Tách ra datagram, chuyển tới nút nhận

- ❑ adaptor là bán tự trị
- ❑ Tầng vật lý và liên kết dữ liệu



# Chương 5: Tầng liên kết

## 5.1 Giới thiệu

## 5.2 Phát hiện lỗi, sửa lỗi

## 5.3 Các giao thức đa truy cập

## 5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

## 5.5 Ảo hóa liên kết: MPLS

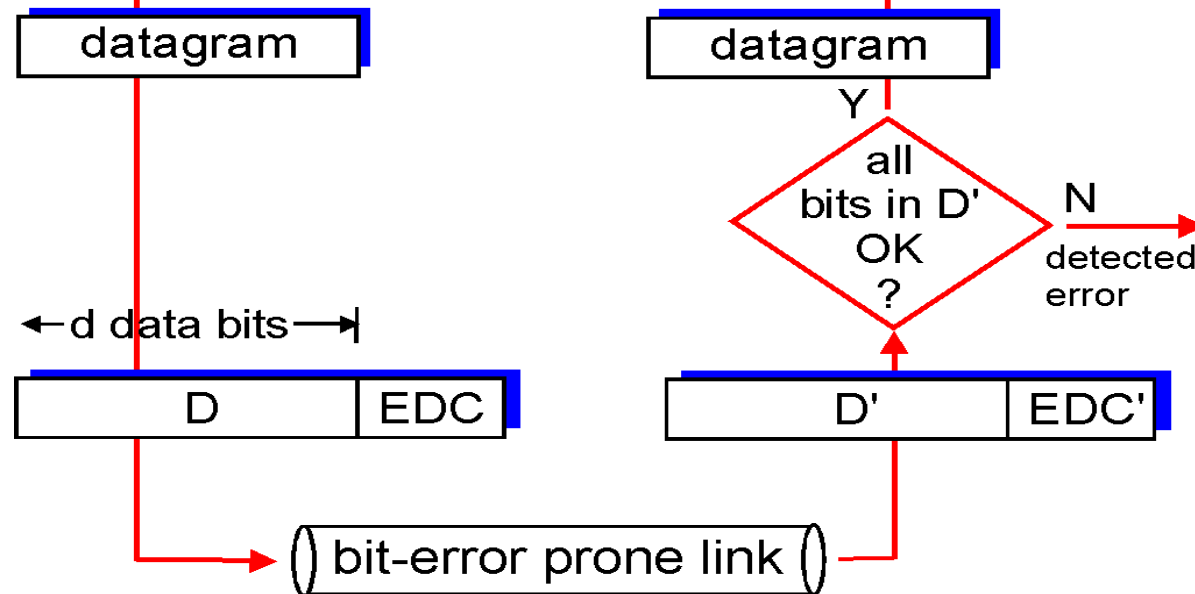
## 5.6 Mạng trung tâm dữ liệu

# Phát hiện lỗi

EDC= Error Detection and Correction bits (phần dôi thừa)

D = Dữ liệu được bảo vệ bằng kiểm tra lỗi, có thể bao gồm các trường header

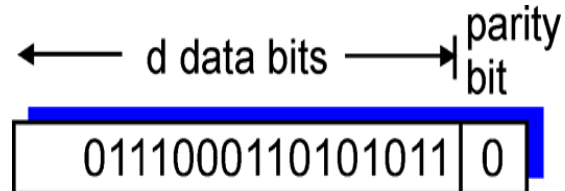
- Phát hiện lỗi không 100% tin cậy!
  - Giao thức có thể không phát hiện ra một số lỗi nhưng tỷ lệ rất thấp
  - Trường EDC càng rộng thì phát hiện và sửa lỗi càng tốt



# Kiểm tra tính chẵn lẻ

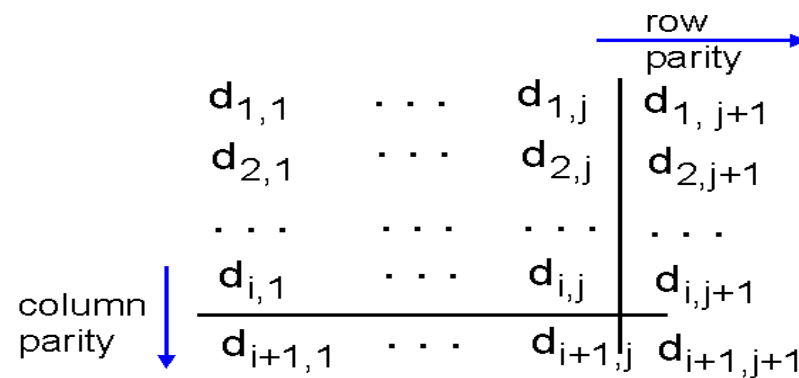
## Bít chẵn lẻ đơn:

Phát hiện một bít lỗi



## Bít chẵn lẻ 2 chiều:

Phát hiện và *sửa* một bít lỗi



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
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
0	0	1	0	1	0

parity error

# Internet checksum

Mục đích: Phát hiện lỗi trong segment đã truyền (Chú ý: *chỉ* sử dụng tại tầng giao vận)

## Bên gửi:

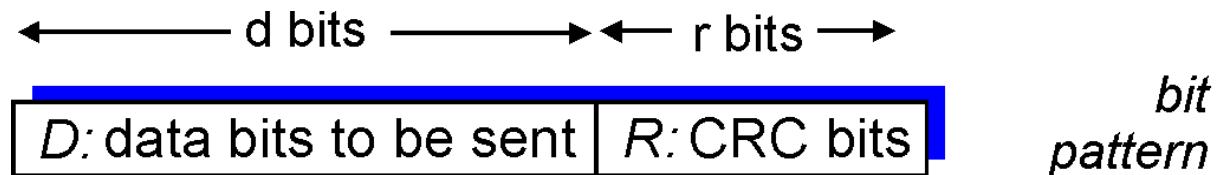
- ❑ Xem xét nội dung của các segment như một chuỗi các số nguyên 16 bit
- ❑ checksum: tổng (bù 1) của nội dung segment
- ❑ Bên gửi đặt giá trị checksum vào trong trường checksum của UDP

## Bên nhận:

- ❑ Tính toán checksum của segment đã nhận
- ❑ Kiểm tra nếu giá trị checksum đã tính bằng giá trị trường checksum:
  - NO – Lỗi được phát hiện
  - YES – Không phát hiện ra lỗi. *Nhưng có thể có lỗi?*

# Checksumming: Cyclic Redundancy Check

- ❑ Coi các bit dữ liệu **D** như một số nhị phân
- ❑ Chọn  $r+1$  bit mẫu (generator), **G**
- ❑ Mục đích: chọn  $r$  bit CRC, **R**, ví dụ
  - $\langle D, R \rangle$  chia hết cho  $G$  (modulo 2)
  - Bên nhận biết  $G$ , chia  $\langle D, R \rangle$  cho  $G$ . Nếu phần dư khác không: lỗi được phát hiện!
  - Có thể phát hiện các lỗi ít hơn  $r+1$  bit
- ❑ Sử dụng rộng rãi trong thực tế (ATM, HDCL)



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

*mathematical formula*

# Ví dụ CRC

Muốn:

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

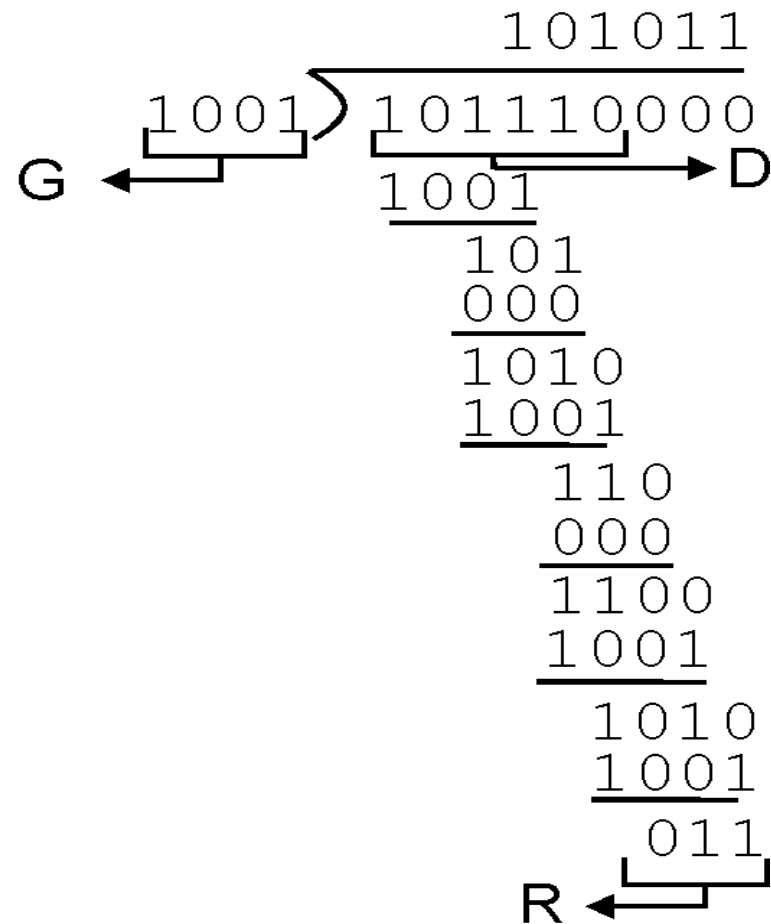
*Tương đương:*

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

*Tương đương:*

Nếu  $D \cdot 2^r$  chia  $G$ ,  
phần dư là  $R$

$$R = \text{Phần dư } \left[ \frac{D \cdot 2^r}{G} \right]$$



# CRC

- ❑ *Chuẩn quốc tế đã định nghĩa 8-, 12-, 16- và 32-bit generator, G.*
- ❑ *8-bit CRC sử dụng để phát hiện lỗi trong ATM cell*
- ❑ *32-bit CRC sử dụng để phát hiện lỗi trong giao thức IEEE của tầng liên kết dữ liệu, sử dụng*

$$G_{CRC-32} = 10000010\ 01100000\ 10001110\ 110110111$$

# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu





# Các giao thức đa truy cập

- ❑ Kênh quảng bá, dùng chung, đơn
- ❑ hai hoặc nhiều nút truyền đồng thời: đan xen
  - **Đụng độ** nếu nút nhận hai hoặc nhiều tín hiệu tại cùng một thời điểm

## Giao thức đa truy cập

- ❑ Thuật toán phân tán xác định cách nút dùng chung kênh (xác định khi nào một nút có thể truyền)
- ❑ Truyền thông về việc dùng chung kênh phải sử dụng chính kênh đó!
  - Không có kênh ở ngoài cùng phối hợp hoạt động

# Ý tưởng của giao thức đa truy cập

## Kênh quảng bá tốc độ $R$ bps

1. Khi một nút muốn truyền, nó có thể gửi với tốc độ  $R$
2. Khi  $M$  nút muốn truyền, mỗi nút có thể gửi với tốc độ trung bình  $R/M$
3. Hoàn toàn không tập trung:
  - Không có nút đặc biệt để phối hợp việc truyền
  - Không có sự đồng bộ của đồng hồ hay khe
4. Đơn giản

# Giao thức MAC: Phân loại

Ba lớp lớn:

## ❑ Phân chia kênh

- Chia kênh thành các phần nhỏ (khe thời gian, tần số, mã)
- Cấp phát phần cho nút sử dụng riêng

## ❑ Truy cập ngẫu nhiên

- Không chia kênh, cho phép đụng độ
- “Khôi phục” từ đụng độ

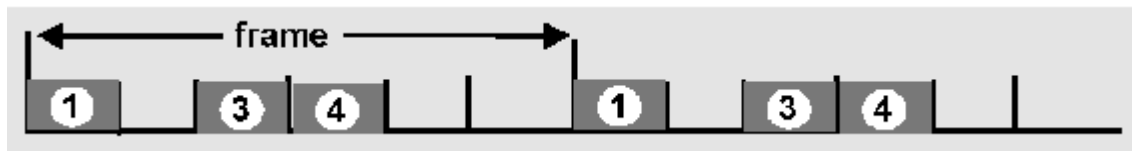
## ❑ Theo phiên lần lượt

- Các nút lần lượt truyền theo phiên nhưng nút cần gửi nhiều có thể chiếm phiên dài hơn

# Giao thức MAC kiểu phân chia kênh: TDMA

## TDMA: time division multiple access

- ❑ Truy cập tới kênh theo vòng
- ❑ Mỗi trạm nhận một khe chiều dài cố định (chiều dài = thời gian truyền gói tin) trong mỗi vòng
- ❑ Các khe không sử dụng sẽ rỗi
- ❑ Ví dụ: 6 trạm LAN, 1,3,4 có gói tin, khe 2,5,6 rỗi

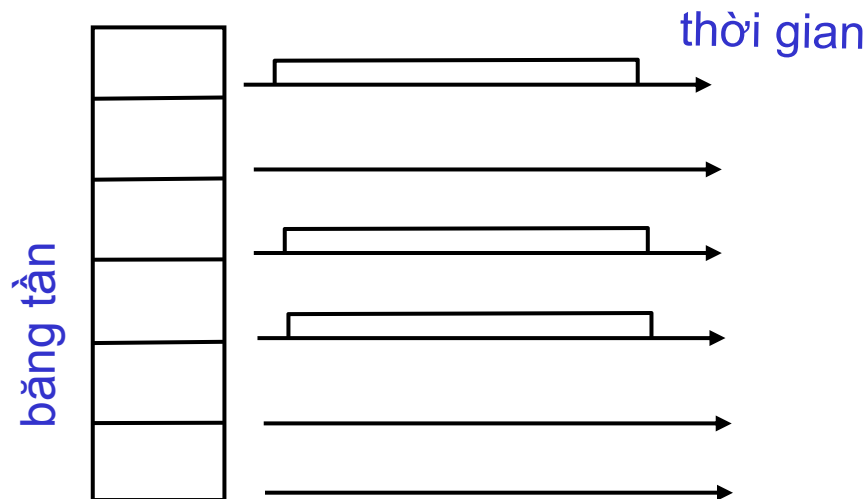


- ❑ TDM (Time Division Multiplexing): kênh được chia thành N khe thời gian, một khe cho một người sử dụng; không hiệu quả với người sử dụng chu kỳ thấp và tải nặng.
- ❑ FDM (Frequency Division Multiplexing): chia theo tần số

# Giao thức MAC kiểu phân chia kênh: FDMA

## FDMA: frequency division multiple access

- ❑ Phổ của kênh được chia thành các băng tần
- ❑ Mỗi trạm được gán một băng tần cố định
- ❑ Thời gian truyền không sử dụng trong băng tần thì sẽ rỗi
- ❑ Ví dụ: 6 trạm LAN; 1,3,4 có gói tin; băng tần 2,5,6 rỗi



- ❑ TDM (Time Division Multiplexing): kênh được chia thành N khe thời gian, một khe cho một người sử dụng; không hiệu quả với người sử dụng chu kỳ thấp và tải nặng
- ❑ FDM (Frequency Division Multiplexing): chia theo tần số

# Giao thức MAC kiểu phân chia kênh: CDMA

# Giao thức truy cập ngẫu nhiên

- ❑ Khi nút có gói tin để gửi
  - Truyền trên toàn kênh với tốc độ dữ liệu  $R$
  - Không có sự phối hợp trước giữa các nút
- ❑ Hai hoặc nhiều nút truyền → “đụng độ”
- ❑ **Giao thức MAC truy cập ngẫu nhiên** chỉ ra:
  - Cách phát hiện đụng độ
  - Cách khôi phục từ đụng độ (ví dụ: đợi một khoảng thời gian rồi truyền lại)
- ❑ Ví dụ của giao thức MAC kiểu truy cập ngẫu nhiên
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA



# Slotted ALOHA

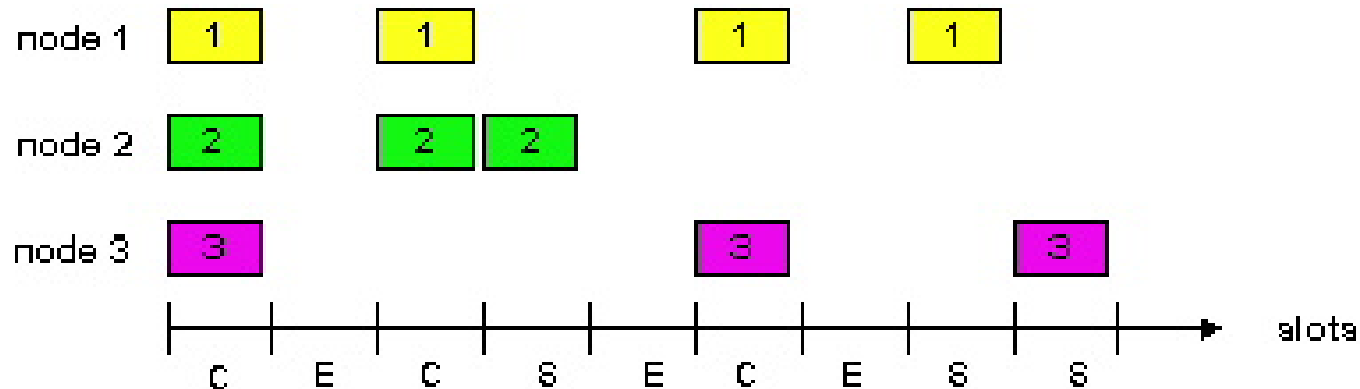
## Giả sử

- ❑ Mọi frame có cùng kích thước
- ❑ Thời gian được chia thành các khe có kích thước bằng nhau, thời gian để truyền 1 frame
- ❑ Nút bắt đầu truyền frame chỉ tại bắt đầu của khe
- ❑ Các nút được đồng bộ
- ❑ Nếu 2 hoặc nhiều nút truyền trong khe, mọi nút phát hiện đụng độ

## Hoạt động

- ❑ Khi nút giành khung mới, nó truyền trong khe tiếp theo
- ❑ Không đụng độ, nút có thể gửi frame mới trong khe tiếp
- ❑ Nếu đụng độ, nút truyền lại frame trong khe sau ngẫu nhiên tới khi thành công

# Slotted ALOHA



## Pros

- ❑ Nút kích hoạt đơn có thể liên tục truyền với tốc độ của cả kênh
- ❑ Không tập chung cao: chỉ các khe trong nút cần được đồng bộ
- ❑ Đơn giản

## Cons

- ❑ Độ trễ, lãng phí khe
- ❑ Các khe rỗi
- ❑ Các nút có thể có khả năng phát hiện độ trễ nhỏ hơn thời gian truyền gói tin
- ❑ Đồng bộ đồng hồ

# Hiệu suất của Slotted Aloha

**Hiệu suất** là phần thời gian dài của các khe thành công khi có nhiều nút, mỗi nút có nhiều frame để gửi

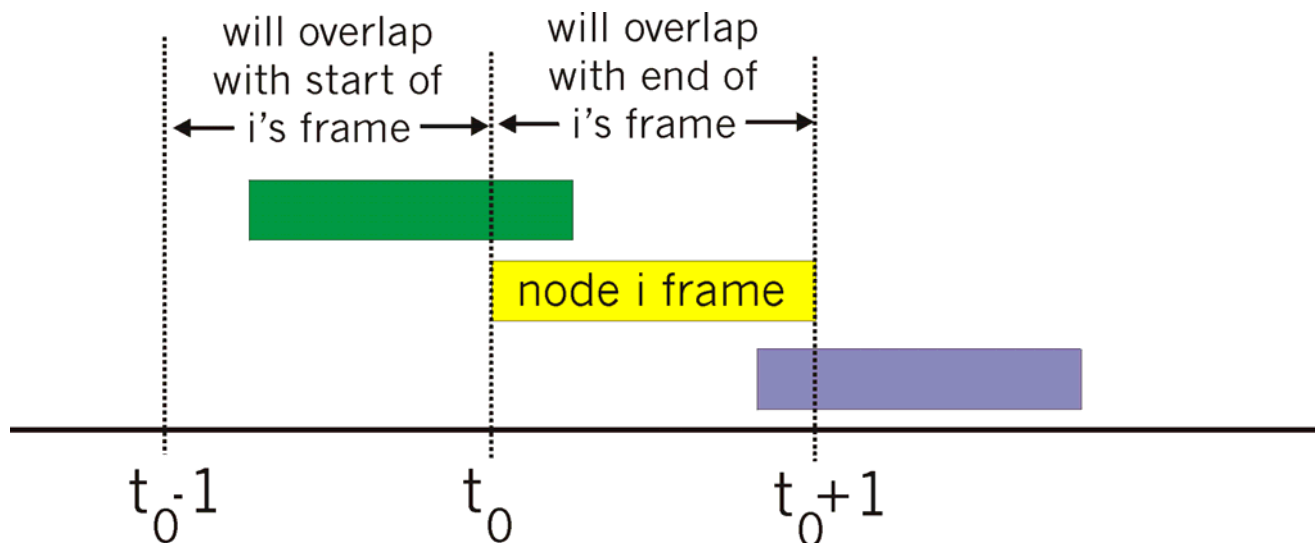
- ❑ Giả sử có  $N$  nút với nhiều frame để gửi, mỗi nút truyền trong khe với xác suất là  $p$
- ❑ Xác suất nút 1 có sự thành công trong một khe =  $p(1-p)^{N-1}$
- ❑ xác suất một nút bất kỳ có sự thành công =  $Np(1-p)^{N-1}$

- ❑ Để tối đa hiệu suất với  $N$  nút, tìm  $p$  để tối đa  $Np(1-p)^{N-1}$
- ❑ Đối với nhiều nút, giới hạn của  $Np^*(1-p^*)^{N-1}$  khi  $N$  tiến ra vô hạn bằng  $1/e = 0.37$

**Tốt nhất:** kênh sử dụng cho việc truyền hữu ích chiếm 37% thời gian

# Pure (unslotted) ALOHA

- ❑ unslotted Aloha: đơn giản, không đồng bộ
- ❑ Khi frame đầu tiên đến
  - truyền ngay lập tức
- ❑ Xác suất đụng độ tăng:
  - frame gửi tại  $t_0$  đụng độ với frame khác gửi trong khoảng  $[t_0-1, t_0+1]$



# Hiệu suất của Pure Aloha

$P(\text{sự thành công của một nút}) = P(\text{nút truyền}) \cdot$

$P(\text{không có nút khác truyền trong } [p_0-1, p_0]) \cdot$

$P(\text{không có nút khác truyền trong } [p_0-1, p_0])$

$$= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$$

$$= p \cdot (1-p)^{2(N-1)}$$

... chọn  $p$  tối ưu và cho  $n$  tiến tới vô hạn ...

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

# CSMA (Carrier Sense Multiple Access)

**CSMA**: Nghe trước khi truyền:

Nếu nghe thấy kênh rỗi: truyền toàn bộ frame

- ❑ Nếu nghe thấy kênh bận, tạm trì hoãn việc truyền

- ❑ Tương tự con người: đừng ngắt lời tôi!

# Đụng độ trong CSMA

Đụng độ có thể vẫn xảy ra:

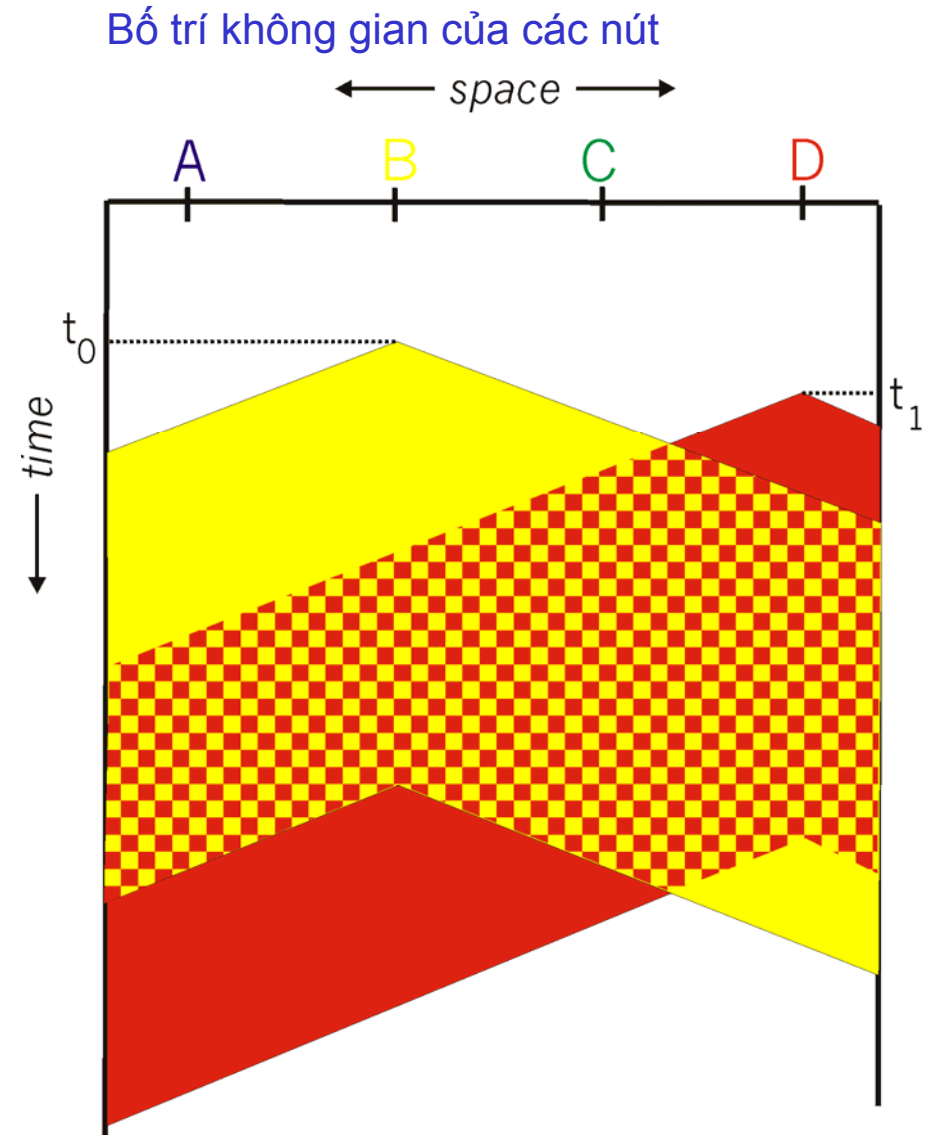
Sự lan truyền độ trễ nghĩa là hai nút có thể không nghe thấy nút khác truyền

Đụng độ:

toàn bộ thời gian gói tin truyền bị lãng phí

Chú ý:

vai trò của khoảng cách và độ trễ lan truyền trong việc xác định xác suất đụng độ



# CSMA/CD (Collision Detection)

**CSMA/CD:** nghe ngóng đường truyền, gọi ngắn như CSMA

- đưng độ *phát hiện* trong thời gian ngắn
- truyền đưng độ bỏ qua, giảm sự lãng phí kênh

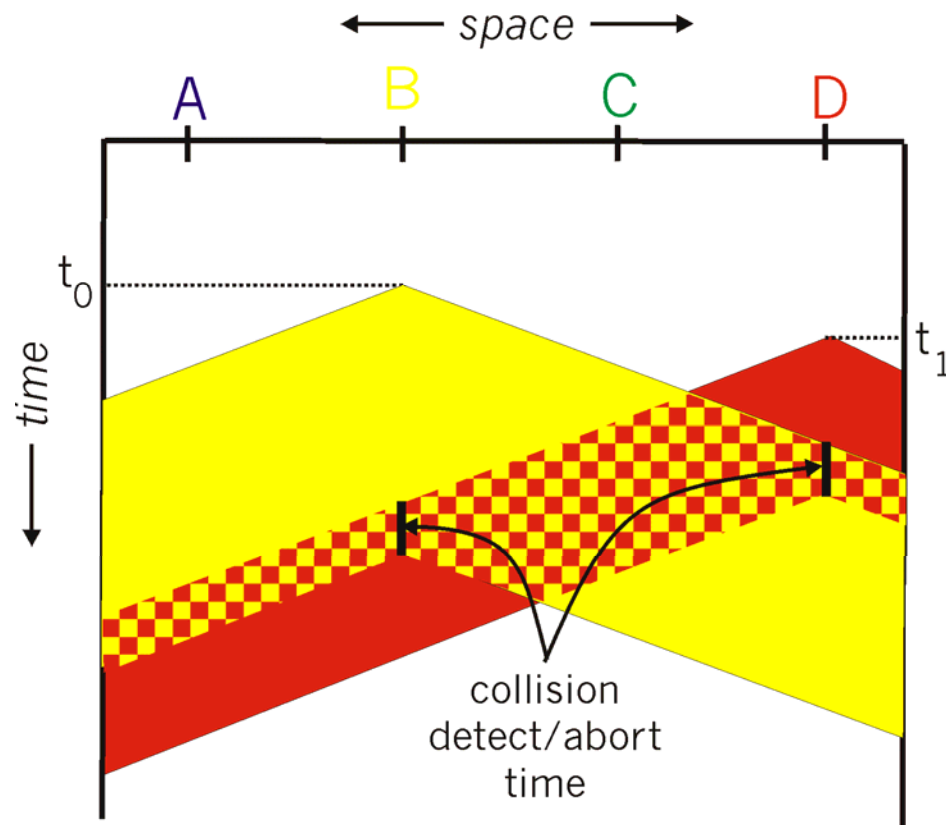
## ❑ Phát hiện đưng độ:

- Dễ trong mạng có dây LAN: đo độ mạnh của tín hiệu, so sánh tín hiệu nhận, truyền
- Khó trong mạng không dây LAN: bên nhận dừng trong khi truyền

## ❑ Tương tự con người: người nói chuyện lịch sự



# Phát hiện đụng độ trong CSMA/CD



# Giao thức MAC “Taking Turns”

## Giao thức MAC kiểu chia kênh:

- Dùng chung kênh hiệu quả và công bằng khi tải nặng
- không hiệu quả khi tải nhẹ: trễ khi truy cập kênh,  $1/N$  bandwidth được cấp thậm chí chỉ có một nút kích hoạt

## Giao thức MAC truy cập ngẫu nhiên

- Hiệu quả khi tải nhẹ: một nút có thể sử dụng toàn bộ kênh
- Tải nặng: độn độ cao

Các giao thức “taking turns”  
tốt cho cả hai trường hợp!

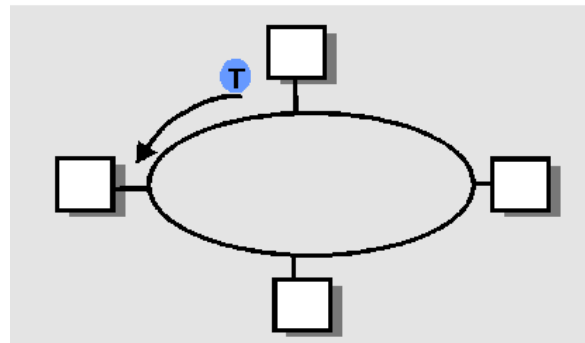
# Giao thức MAC “Taking Turns”

## Thăm dò:

- ❑ Nút chủ mời nút tớ truyền theo thứ tự
- ❑ Xem xét:
  - overhead của thăm dò
  - trễ
  - một điểm lỗi (nút chủ)

## Chuyển thẻ bài:

- ❑ Điều khiển **thẻ bài** chuyển từ một nút tới nút tiếp theo thứ tự.
- ❑ Bản tin thẻ bài
- ❑ Xem xét:
  - Overhead của thẻ bài
  - Độ trễ
  - Một điểm lỗi (thẻ bài)



# Tổng kết về giao thức MAC

- ❑ Chúng ta làm gì với phương tiện dùng chung?
  - Chia kênh theo thời gian, tần số, mã
    - Time Division, Frequency Division
  - Chia ngẫu nhiên (động)
    - ALOHA, S-ALOHA, CSMA, CSMA/CD
    - Nghe ngóng đường truyền: dễ trong một số công nghệ (có dây), khó trong một số công nghệ (không dây)
    - CSMA/CD sử dụng trong Ethernet
    - CSMA/CA sử dụng trong 802.11
  - Taking Turns
    - Thăm dò từ trung tâm, chuyển thẻ bài

# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu

# MAC và ARP

## ❑ 32-bit địa chỉ IP:

- Địa chỉ của tầng mạng cho interface
- sử dụng cho tầng 3 (tầng mạng) để chuyển tiếp

## ❑ Địa chỉ MAC (hoặc địa chỉ LAN, địa chỉ vật lý, địa chỉ Ethernet):

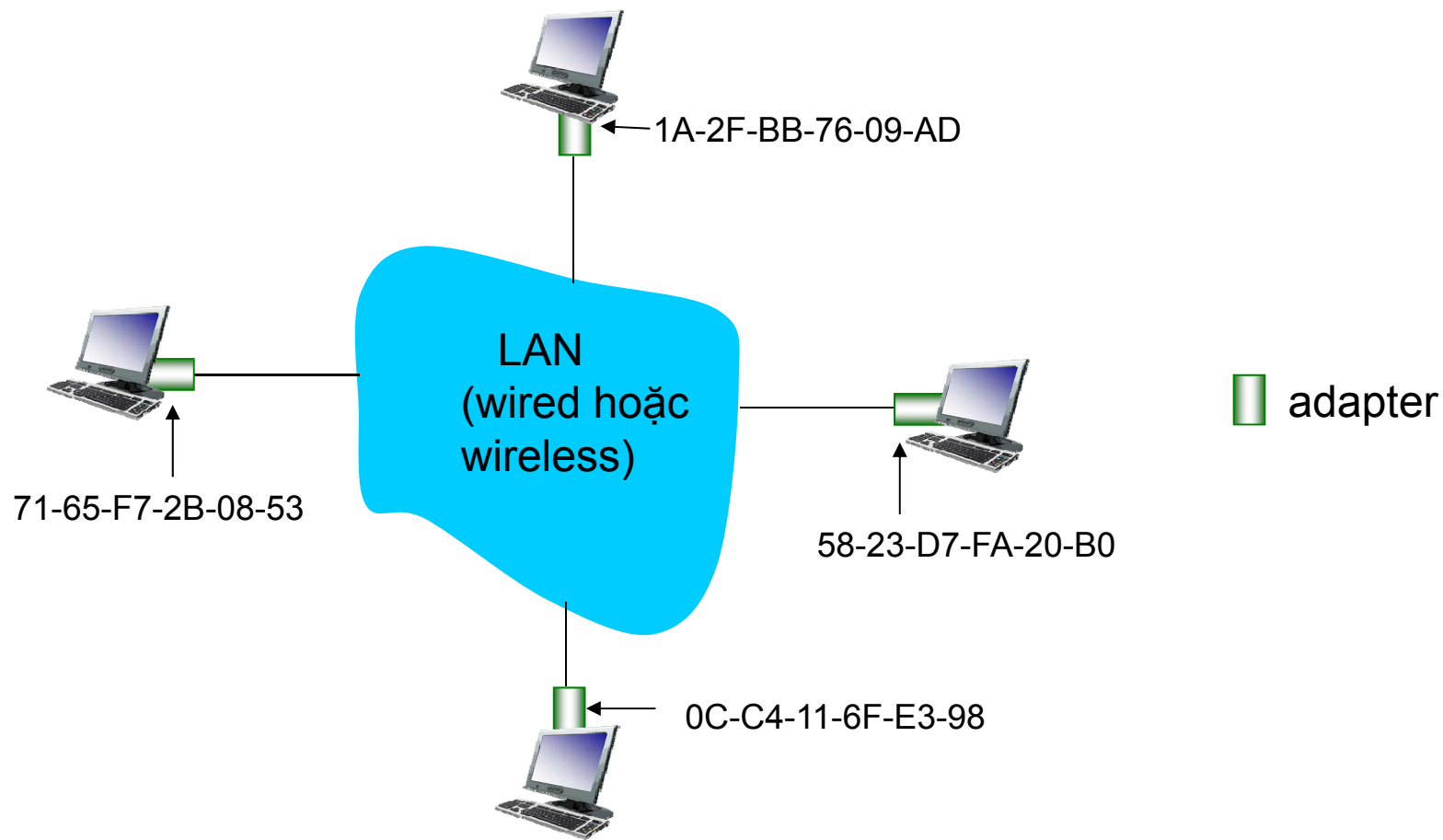
- chức năng: *used 'locally' to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)*
- 48 bit địa chỉ MAC gắn mới mỗi NIC (Network interface controller / network interface card / LAN adapter / physical network interface)

- ví dụ: 1A-2F-BB-76-09-AD

biểu diễn ở hệ 16, mỗi ký tự là giá trị của 4 bit

# Địa chỉ LAN và ARP

Mỗi adapter trong LAN có 1 địa chỉ LAN duy nhất



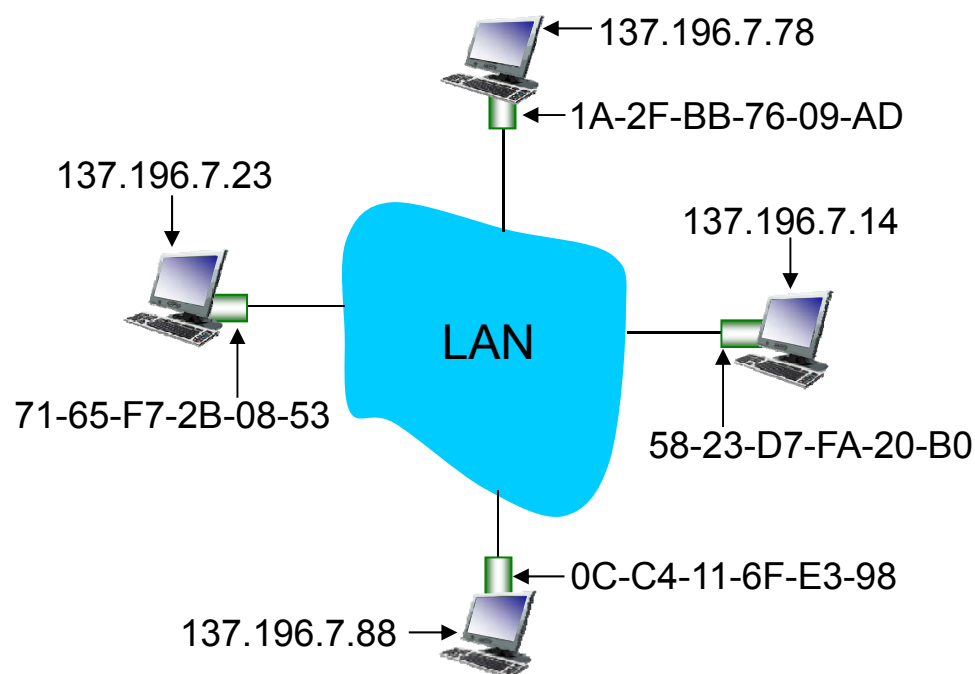
# Địa chỉ LAN

- ❑ IEEE quản lý việc cấp phát địa chỉ MAC
- ❑ Các nhà sản xuất mua không gian địa chỉ MAC (để đảm bảo duy nhất)
- ❑ Địa chỉ MAC phẳng (flat address): có tính di động
  - có thể mang LAN card từ LAN này sang LAN khác
- ❑ Địa chỉ IP phân cấp: không có tính di động
  - địa chỉ phụ thuộc vào IP subnet mà nút mạng kết nối vào



# ARP: address resolution protocol

Xác định địa chỉ MAC của interface như thế nào khi biết địa chỉ IP của nó?



*ARP table*: mỗi IP node (host, router) trong LAN có bảng

- ánh xạ địa chỉ IP/MAC của một số nút trong LAN:  
< IP address; MAC address; TTL >
- TTL (Time To Live): thời gian sau đó ánh xạ địa chỉ sẽ bị xóa (thường 20 phút)

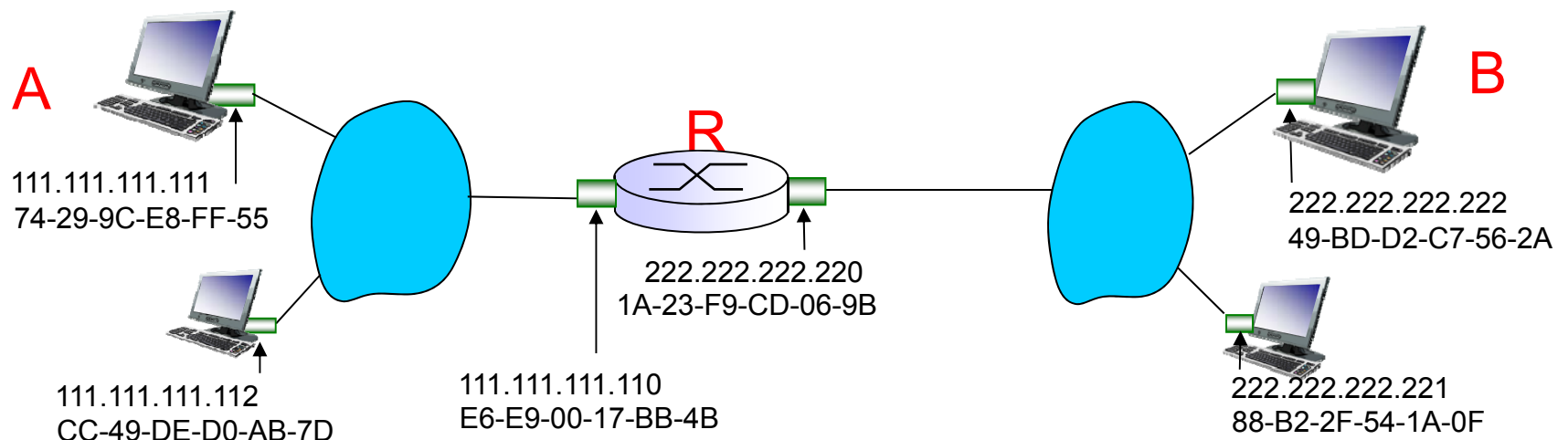
# Giao thức ARP: cùng LAN

- ❑ A muốn gửi datagram cho B
  - Địa chỉ MAC của B không có trong ARP table của A
- ❑ A **quảng bá (broadcast)** ARP query packet, chứa địa chỉ IP của B
  - dest MAC address = FF-FF-FF-FF-FF-FF
  - tất cả mọi nút trong LAN sẽ nhận ARP query packet
- ❑ B nhận ARP packet, trả lời A với địa chỉ MAC của B
  - frame gửi tới địa chỉ MAC của A (unicast)
- ❑ A lưu giữ cặp địa chỉ IP-to-MAC trong ARP table trong một khoảng thời gian
- ❑ ARP thực hiện “plug-and-play”:
  - các nút tạo ARP table không có sự can thiệp của quản trị mạng

# Addressing: Dẫn đường tới LAN khác

xem xét: **gửi datagram từ A tới B qua R**

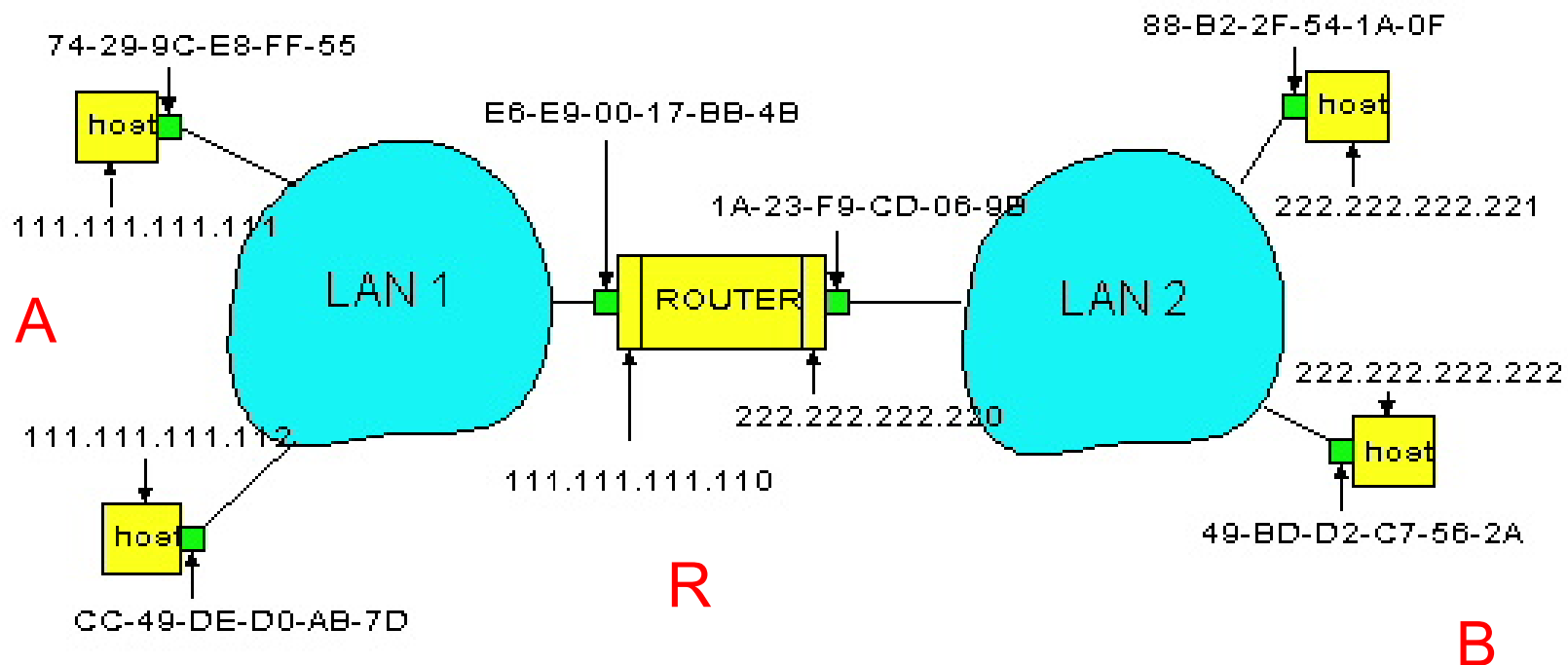
- vấn đề địa chỉ tại tầng IP (datagram) và tầng MAC (frame)
- giả sử A biết địa chỉ IP của B
- giả sử A biết địa chỉ IP của router kề, R (làm sao A biết được?)
- giả sử A biết địa chỉ MAC của R (làm sao A biết được?)



# Dẫn đường tới mạng LAN khác

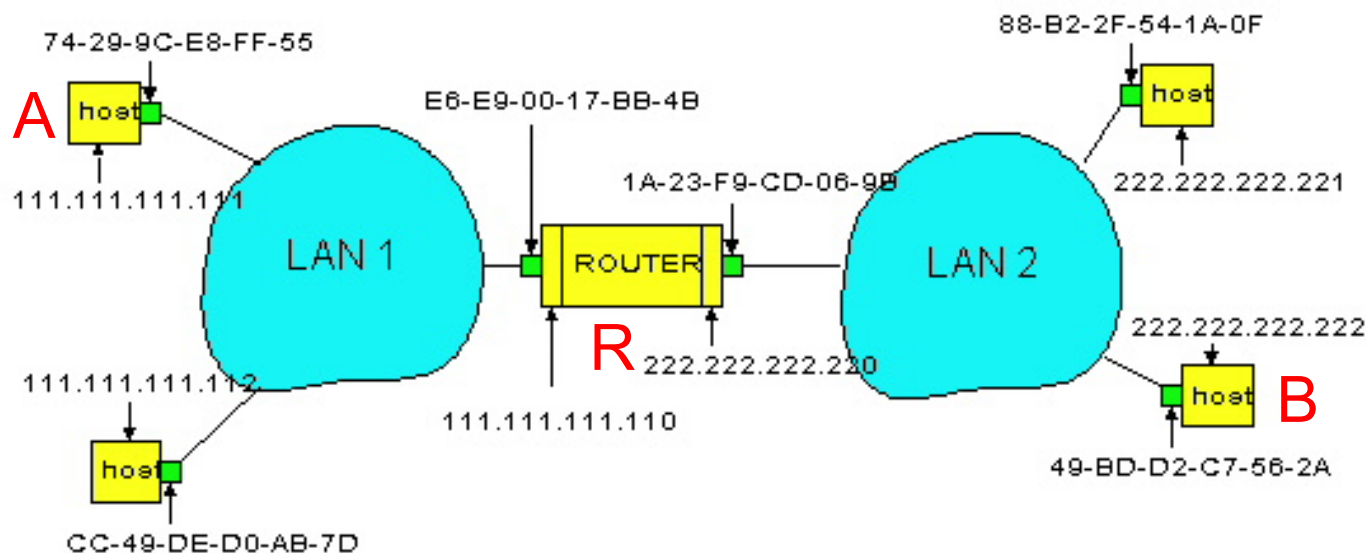
Gửi datagram từ A tới B qua R

Giả sử A biết địa chỉ IP của B



- ❑ Hai bảng ARP trong router R, mỗi bảng cho một mạng LAN
- ❑ Trong bảng dẫn đường tại host nguồn, tìm router 111.111.111.110
- ❑ Trong bảng ARP tại nguồn, tìm địa chỉ MAC E6-E9-00-17-BB-4B,...

- ❑ A tạo datagram có địa chỉ nguồn A, địa chỉ đích B
- ❑ A sử dụng ARP để biết địa chỉ MAC của R 111.111.111.110
- ❑ A tạo frame tầng liên kết dữ liệu, sử dụng địa chỉ MAC của R là địa chỉ đích, frame chứa IP datagram A-to-B
- ❑ Adapter của A gửi frame
- ❑ Adapter của R nhận frame
- ❑ R lấy IP datagram từ Ethernet frame, thấy địa chỉ đích là tới B
- ❑ R sử dụng ARP để có địa chỉ MAC của B
- ❑ R tạo frame chứa IP datagram A-to-B gửi tới B



# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

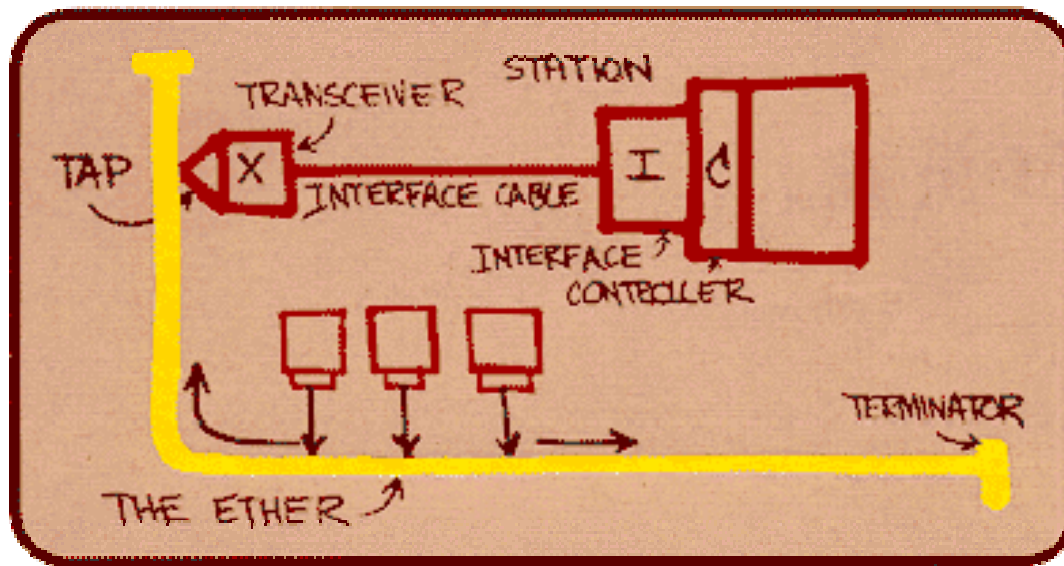
5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu

# Ethernet

Công nghệ LAN có dây phổ biến:

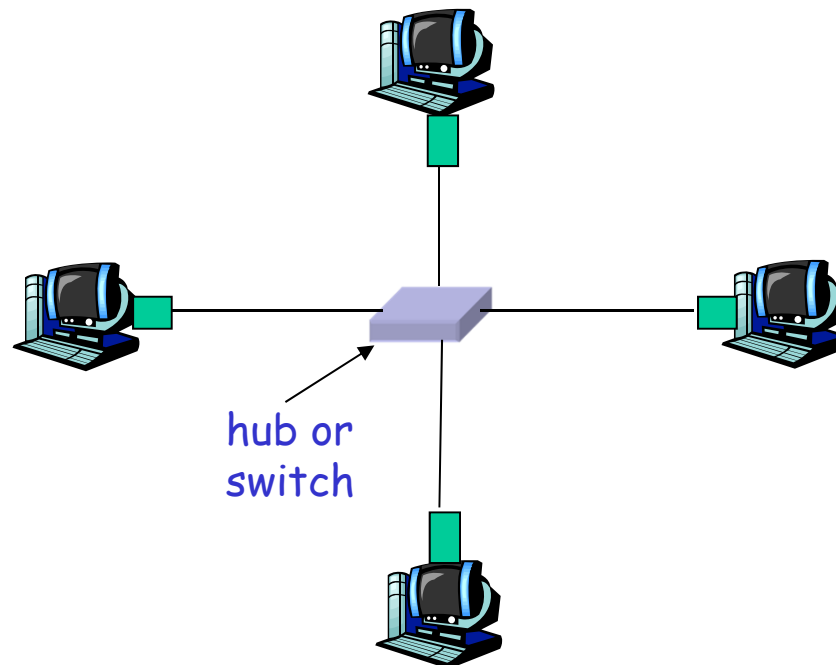
- ❑ Rẻ \$20 với 100Mbps!
- ❑ Công nghệ LAN đầu tiên sử dụng rộng rãi
- ❑ Đơn giản, rẻ hơn token LAN và ATM
- ❑ Có sự cạnh tranh về tốc độ : 10 Mbps – 10 Gbps



Phác thảo Ethernet của Metcalfe

# Star topology

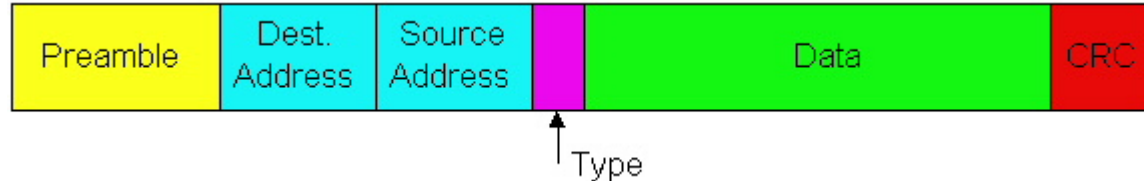
- ❑ Bus topology phổ biến vào giữa những năm 90
- ❑ Hiện nay, star topology phổ biến
- ❑ Sự lựa chọn kết nối: hub hay switch





# Cấu trúc của Ethernet Frame

Adapter bên gửi đóng gói IP datagram (hoặc các gói tin tầng mạng khác) trong **Ethernet frame**

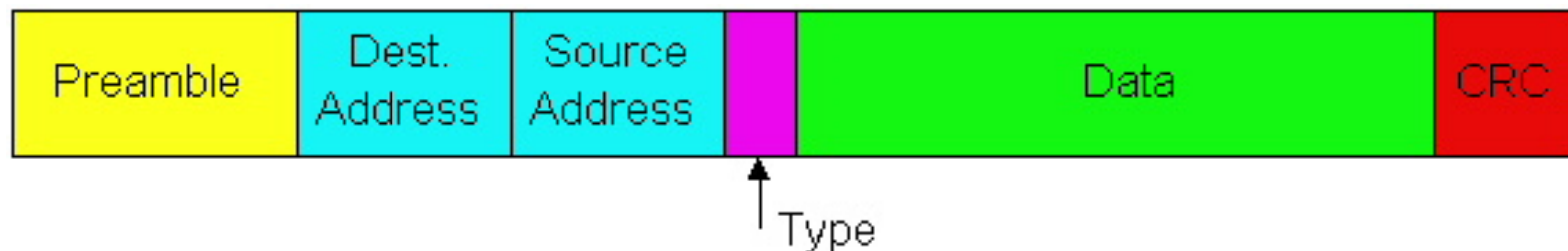


## Preamble:

- ❑ 7 bytes with pattern 10101010 theo sau là 1 byte với pattern 10101011
- ❑ Sử dụng để đồng bộ tốc độ bên nhận, bên gửi

# Cấu trúc của Ethernet Frame

- ❑ **Address:** 6 byte
  - Nếu adapter nhận frame có địa chỉ đích tương ứng hoặc với địa chỉ quảng bá (ví dụ: gói tin ARP), nó chuyển dữ liệu trong frame tới giao thức tầng mạng
  - Trường hợp khác, adapter loại bỏ frame
- ❑ **Type:** chỉ ra giao thức của tầng phía trên (hầu hết là IP nhưng cũng có thể là Novell IPX hay AppleTalk)
- ❑ **CRC:** được kiểm tra tại bên nhận, nếu phát hiện ra lỗi thì frame bị loại bỏ



# Dịch vụ không hướng kết nối, không tin cậy

- ❑ **Không hướng kết nối:** Không bắt tay giữa adapter bên gửi và adapter bên nhận
- ❑ **Không tin cậy:** adapter bên nhận không gửi ack hoặc nack cho adapter bên gửi
  - Luồng datagram chuyển tới tầng mạng có thể khe hở
  - Các khe hở sẽ được điền đầy nếu ứng dụng sử dụng TCP
  - Ngược lại, ứng dụng sẽ thấy các khe hở

# Ethernet sử dụng CSMA/CD

- ❑ Không slot
- ❑ Adapter không truyền nếu nó nghe thấy một số adapter đang truyền, gọi là **nghe ngóng đường truyền (carrier sense)**
- ❑ Adapter bỏ qua truyền khi nó nghe thấy adapter khác đang truyền, gọi là **phát hiện đụng độ (collision detection)**
- ❑ Trước khi cố gắng truyền lại, adapter đợi một thời gian ngẫu nhiên, gọi là **truy cập ngẫu nhiên (random access)**

# Thuật toán Ethernet CSMA/CD

1. Adaptor nhận datagram từ tầng mạng và tạo frame
2. Nếu adapter nghe ngóng thấy kênh rỗi, nó bắt đầu truyền frame. Nếu nó nghe ngóng thấy kênh bận, đợi tới khi kênh rỗi và truyền
3. Nếu adapter truyền toàn bộ frame không phát hiện thấy việc truyền khác, adapter đã xong với frame !
4. Nếu adapter phát hiện việc truyền khác trong khi truyền, bỏ qua và gửi tín hiệu jam
5. Sau khi bỏ qua, adapter vào **exponential backoff**: sau đựng độ thứ m, adapter chọn K ngẫu nhiên từ  $\{0, 1, 2, \dots, 2^m - 1\}$ . Adapter đợi  $K \cdot 512$  bit thời gian và quay lại bước 2

# Ethernet CSMA/CD

**Jam Signal:** đảm bảo chắc chắn mọi bộ phát biết về đụng độ; 48 bit

**Bit time:** .1 microsec cho 10 Mbps Ethernet; với  $K=1023$ , thời gian đợi là khoảng 50 msec

## Exponential Backoff:

- ❑ *Mục đích:* thích nghi việc thử truyền lại, ước lượng tải hiện tại
  - tải nặng: đợi ngẫu nhiên sẽ lâu hơn
- ❑ Đụng độ đầu tiên: chọn  $K$  từ  $\{0,1\}$ ; độ trễ là  $K \cdot 512$  bit thời gian truyền
- ❑ Sau đụng độ thứ hai: chọn  $K$  từ  $\{0,1,2,3\}...$
- ❑ Sau 10 đụng độ, chọn  $K$  từ  $\{0,1,2,3,4,...,1023\}$

# Hiệu suất của CSMA/CD

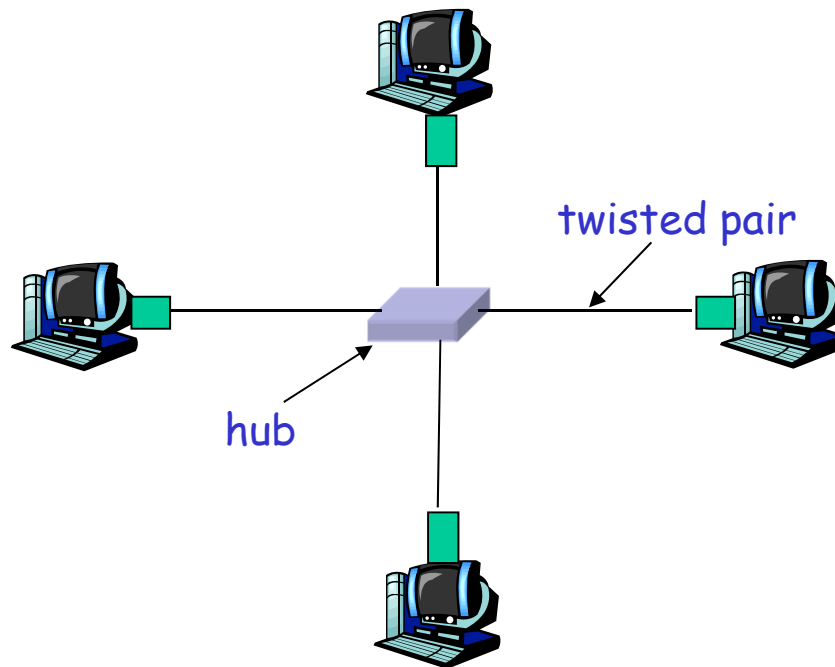
- $T_{prop}$  = xác suất lớn nhất giữa 2 nút trong LAN
- $t_{trans}$  = thời gian để truyền frame có kích thước lớn nhất

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

- Hiệu suất tới 1 khi  $t_{prop}$  tiến tới 0
- Tới 1 khi  $t_{trans}$  tiến tới vô hạn
- Tốt hơn nhiều ALOHA, nhưng vẫn không tập chung, đơn giản và rẻ

# 10BaseT và 100BaseT

- ❑ Tốc độ 10/100 Mbps; 100 Mbps gọi là “fast ethernet”
- ❑ T viết tắt của Twisted Pair
- ❑ Các nút kết nối tới hub: “star topology”; khoảng cách giữa nút và hub tối đa 100m

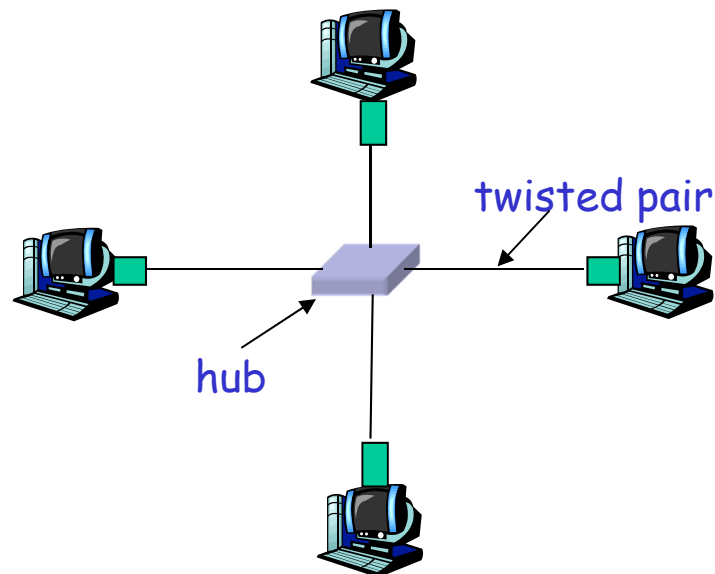




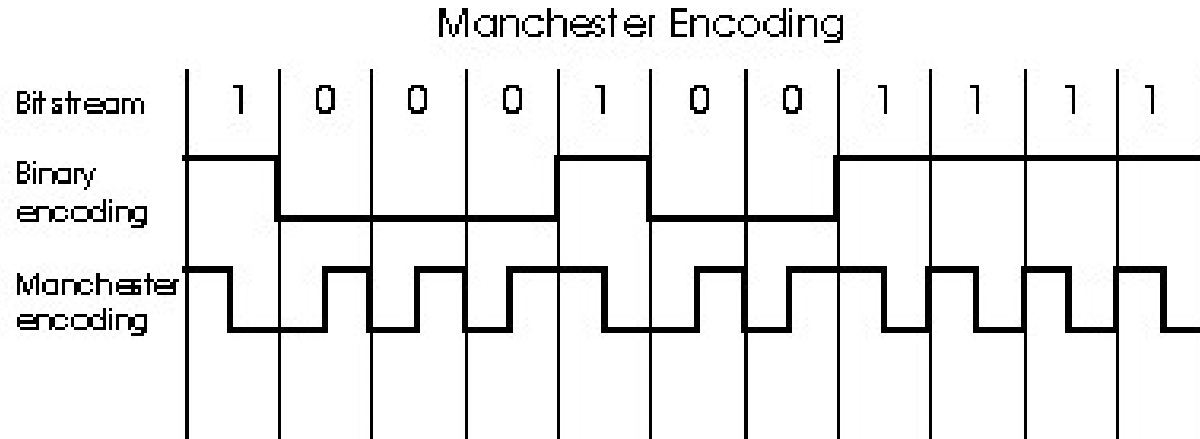
# Hub

Hub là thiết bị lặp của tầng vật lý:

- Các bit đến từ một liên kết sẽ ra mọi liên kết khác
- Với cùng tốc độ
- Không có vùng đệm frame
- Không có CSMA/CD tại hub: adapter phát hiện đụng độ
- Cung cấp chức năng quản trị mạng



# Mã Manchester



- ❑ Sử dụng 10BaseT
- ❑ Mỗi bit có một transition
- ❑ Cho phép đồng hồ trong nút gửi và nhận để đồng bộ tại nút kia
  - Không cần có trung tâm, đồng hồ chung giữa các nút!
- ❑ Xét ở tầng vật lý!

# Gbit Ethernet

- ❑ Sử dụng định dạng Ethernet frame chuẩn
- ❑ Cho phép liên kết point-to-point và kênh quảng bá dùng chung
- ❑ Trong chế độ dùng chung, sử dụng CSMA/CD; giữa các nút đòi hỏi khoảng cách ngắn để có hiệu quả
- ❑ Sử dụng hub, gọi là “Buffered Distributors”
- ❑ Full-Duplex tại 1 Gbps cho liên kết point-to-point
- ❑ Hiện tại, 10 Gbps!

# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

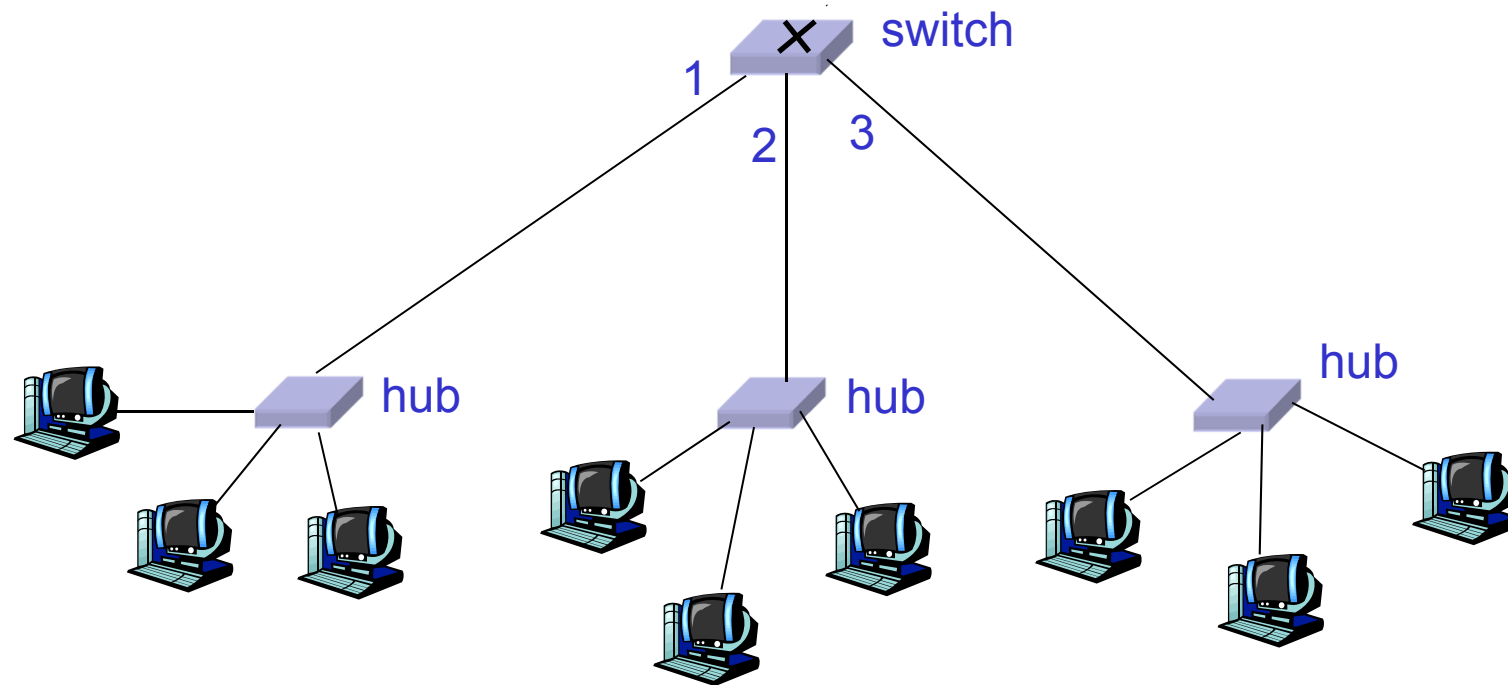
5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu

# Switch

- ❑ Thiết bị thực hiện tầng liên kết dữ liệu
  - Lưu và chuyển tiếp Ethernet frames
  - Kiểm tra frame header và chọn chuyển tiếp frame dựa trên địa chỉ đích MAC
  - Khi frame được chuyển tiếp trên segment, sử dụng CSMA/CD để truy cập segment
- ❑ Trong suốt
  - Host không biết về sự tồn tại của switch
- ❑ plug-and-play, tự học
  - Switch không cần phải cấu hình

# Chuyển tiếp



- Cách xác định chuyển tiếp frame đến LAN segment nào?
- Giống bài toán dẫn đường...

# Tự học

- ❑ Switch có **bảng switch**
- ❑ Entry trong bảng switch:
  - (MAC Address, Interface, Time Stamp)
  - Các entry cũ trong bảng sẽ bị xóa đi (TTL có thể 60 phút)
- ❑ Switch **học** host nào có thể tới qua interface nào
  - Khi nhận frame, switch học vị trí của địa chỉ gửi: incoming LAN segment
  - Ghi cặp địa chỉ gửi/vị trí trong bảng switch

# Lọc/Chuyển tiếp

Khi switch nhận một frame:

Chỉ số bảng switch sử dụng địa chỉ MAC đích

**if** thấy entry cho đích

**then{**

**if** đích trong segment mà frame đến

**then** Loại bỏ frame

**else** Chuyển tiếp frame trên interface chỉ định

**}**

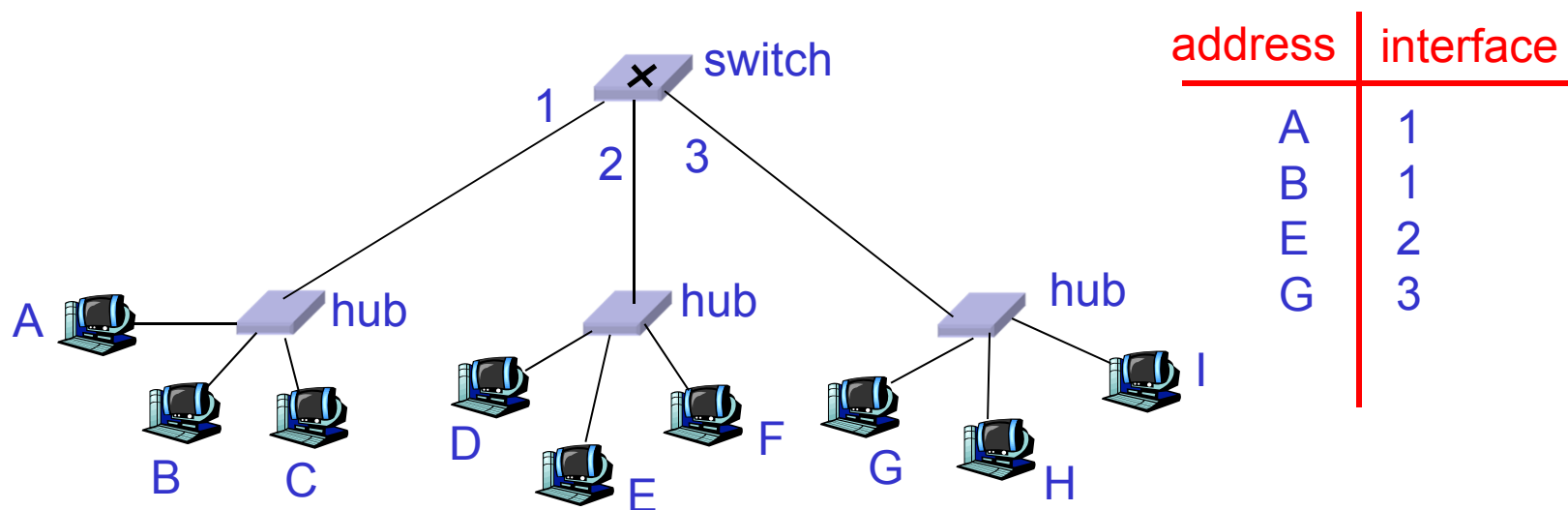
**else** Làm tràn

*Chuyển tiếp trên tất cả interface trừ interface mà frame đã tới*



# Ví dụ Switch

Giả sử C gửi frame tới D



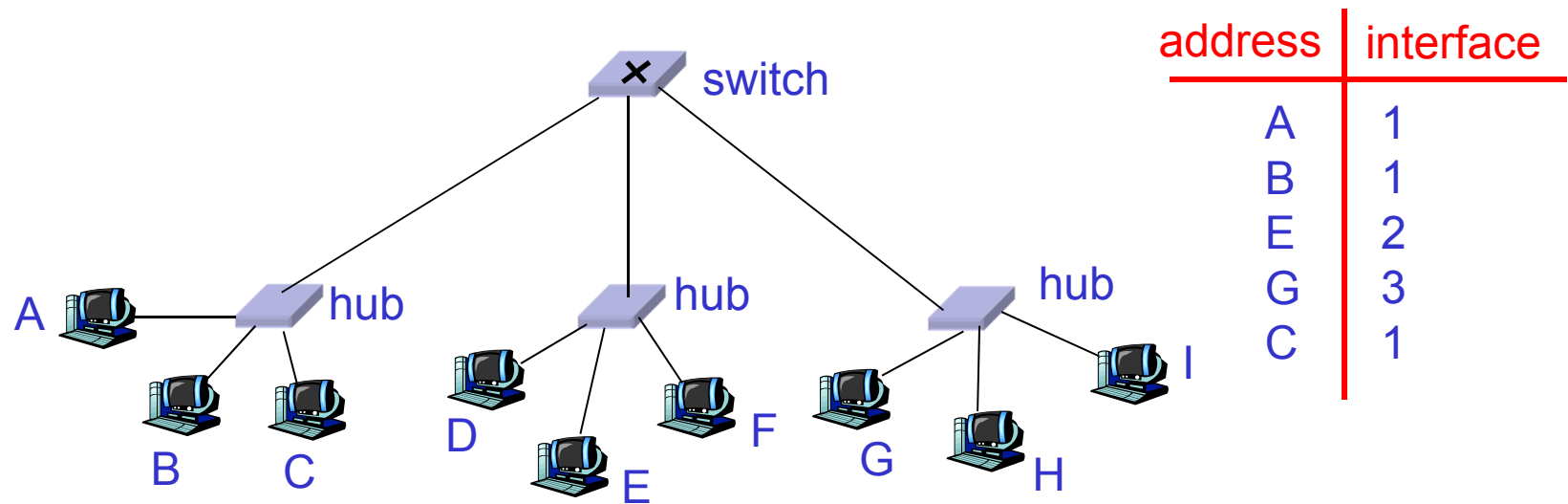
## ❑ Switch nhận frame từ C

- Ghi vào bảng switch là C trên interface 1
- Bởi vì D không trong bảng, switch chuyển tiếp frame qua interface 2 và 3

## ❑ frame được nhận bởi D

# Ví dụ Switch

Giả sử D gửi frame trả lời C



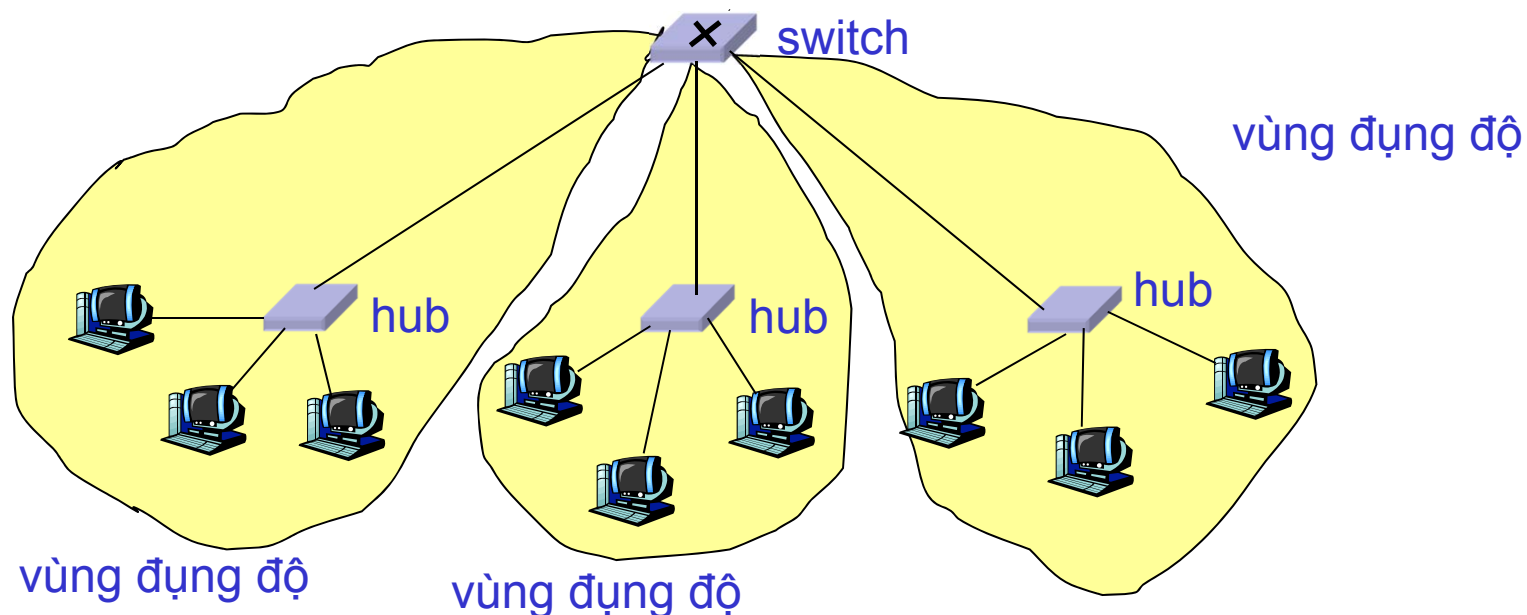
## ❑ Switch nhận frame từ D

- Ghi vào bảng switch là D ở trên interface 2
- Bởi vì C có trong bảng, switch chuyển tiếp frame chỉ tới interface 1

## ❑ frame được nhận bởi C

# Switch: cô lập lưu lượng

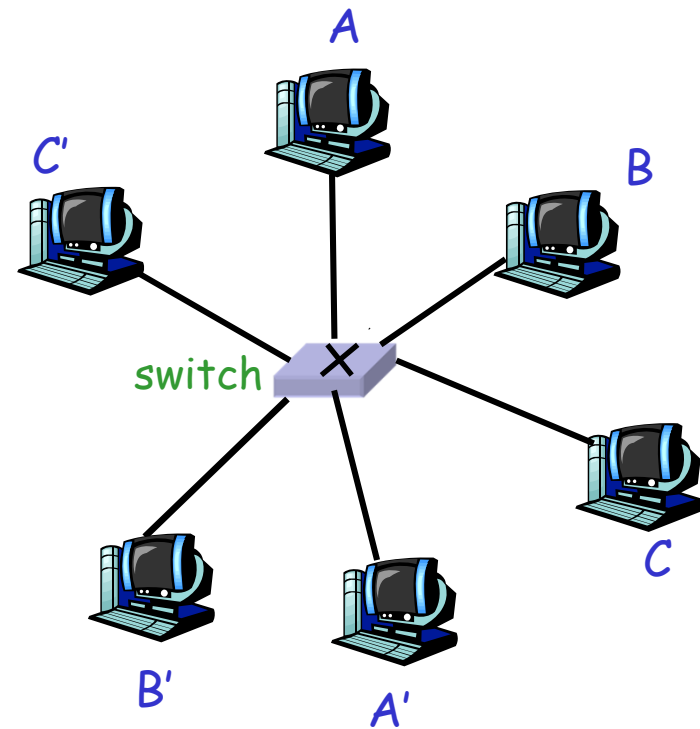
- ❑ Cài đặt switch chia subnet thành các LAN segment
- ❑ Switch **lọc** gói tin:
  - Frame cùng LAN-segment thường không chuyển tiếp trên LAN segment khác
  - Segment được phân tách các **vùng độn độ**



# Switch: truy cập dành riêng

- ❑ Switch có nhiều interface
- ❑ Các host có kết nối trực tiếp tới switch
- ❑ Không đụng độ; full duplex

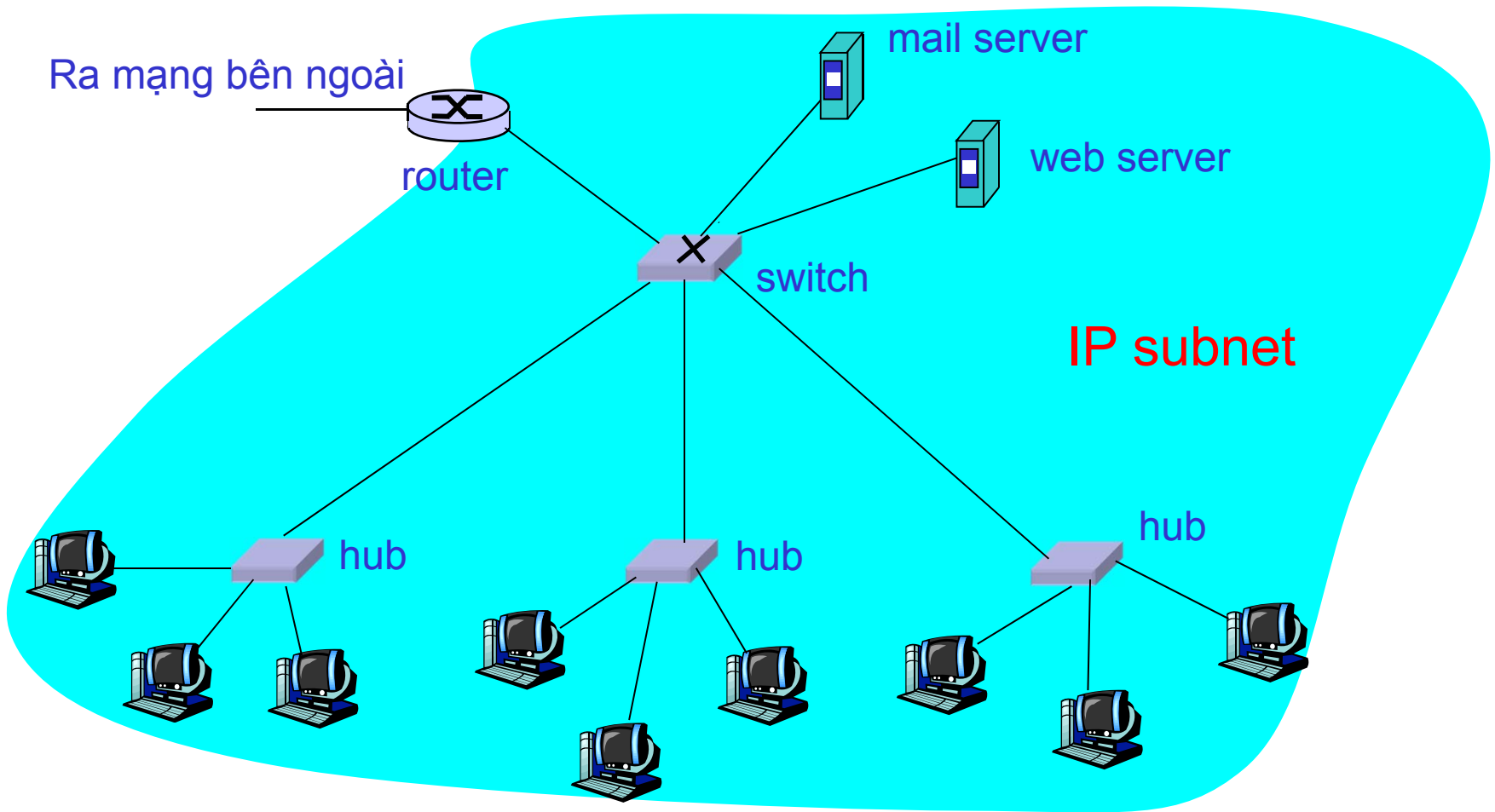
**Switching:** A-to-A' và B-to-B' đồng thời, không đụng độ



# Một số thông tin khác về Switch

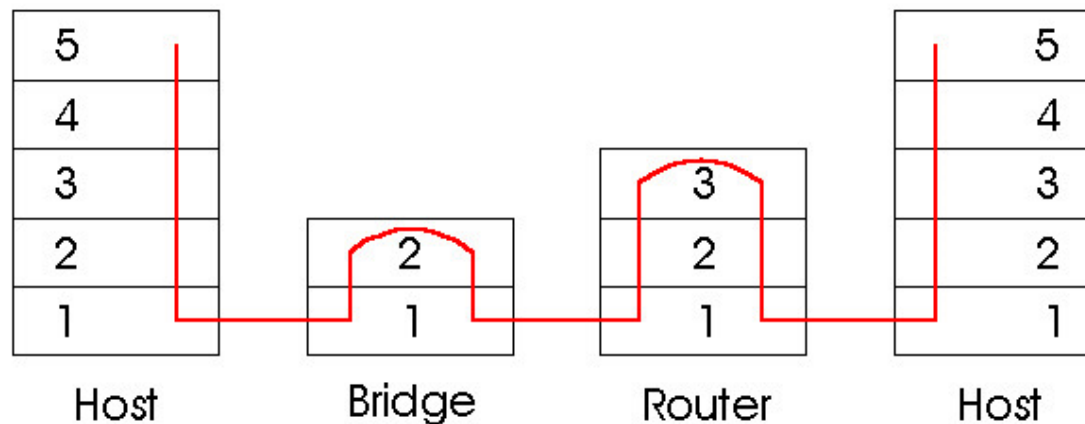
- ❑ **cut-through switching:** frame chuyển tiếp từ input port tới output port không chờ nhận đủ frame
  - Giảm một chút về độ trễ (latency)
- ❑ Sự kết của dùng chung/dành riêng, 10/100/1000 Mbps interface

# Mạng của một tổ chức



# So sánh Switch và Router

- ❑ Các thiết bị store-and-forward
  - router: Thiết bị tầng mạng (kiểm tra header của tầng mạng)
  - switch: Thiết bị tầng liên kết dữ liệu
- ❑ Routers duy trì bảng dẫn đường, thực hiện thuật toán dẫn đường
- ❑ Switch duy trì bảng switch, thực hiện thuật toán lọc, học



# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

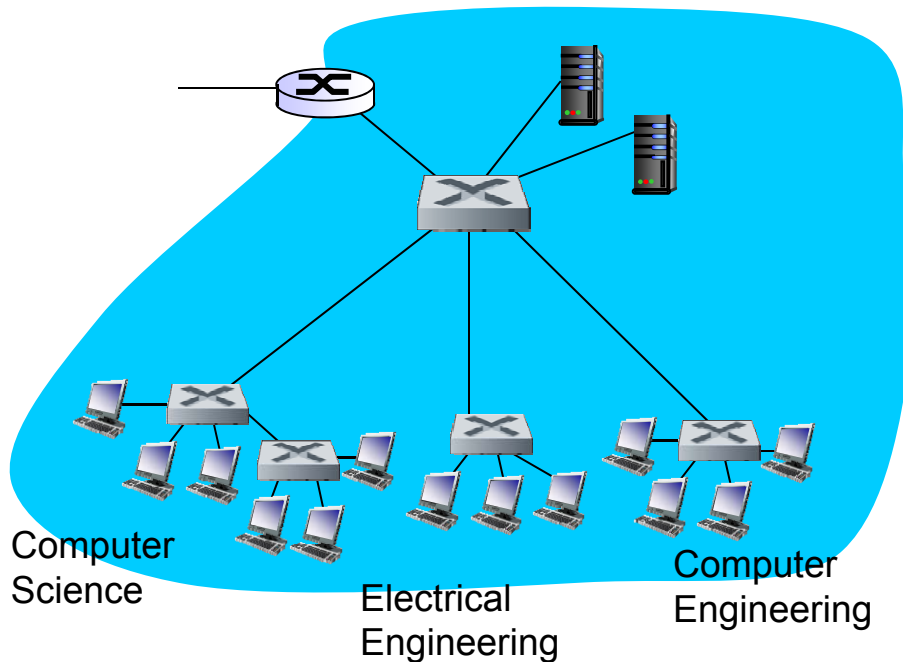
- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu



# Mục đích VLAN



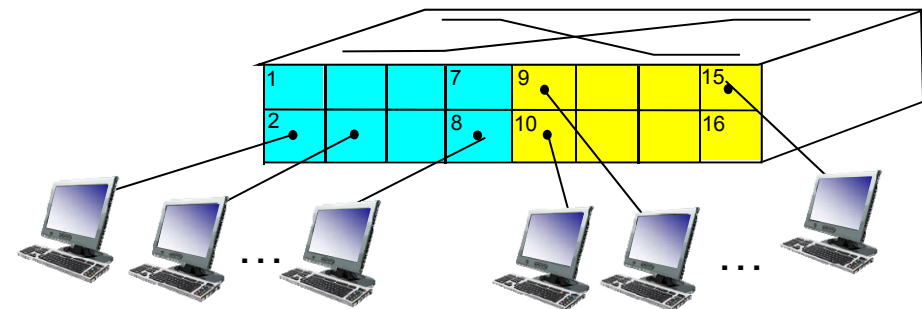
- ❑ Người của CS chuyển văn phòng tới EE, nhưng muốn kết nối tới switch của CS?
- ❑ Một vùng quảng bá duy nhất:
  - tất cả layer-2 quảng bá lưu lượng (ARP, DHCP, vị trí không biết của địa chỉ MAC đích) phải qua toàn bộ LAN
  - vấn đề an toàn bảo mật, vấn đề hiệu quả

# VLAN

## *Virtual Local Area Network*

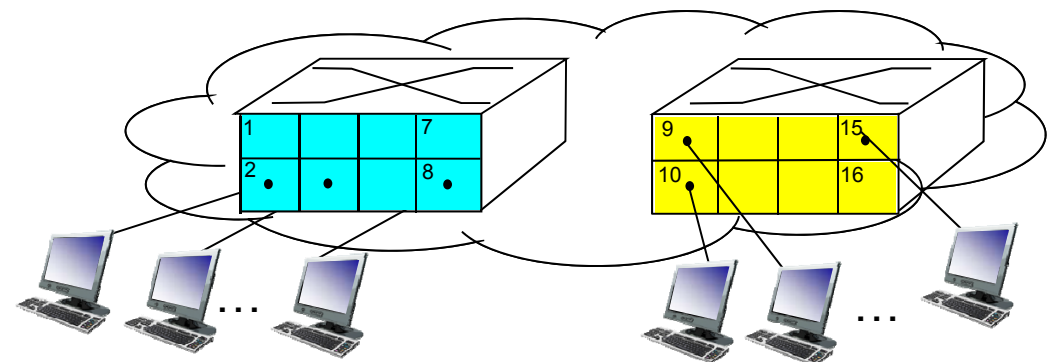
Các switch hỗ trợ VLAN có thể được cấu hình để xác định nhiều mạng LAN ảo trên một hạ tầng LAN vật lý

**port-based VLAN:** các switch port được nhóm lại (bằng phần mềm quản lý switch) để một switch vật lý duy nhất hoạt động như nhiều switch ảo



Electrical Engineering  
(VLAN ports 1-8)

Computer Science  
(VLAN ports 9-15)

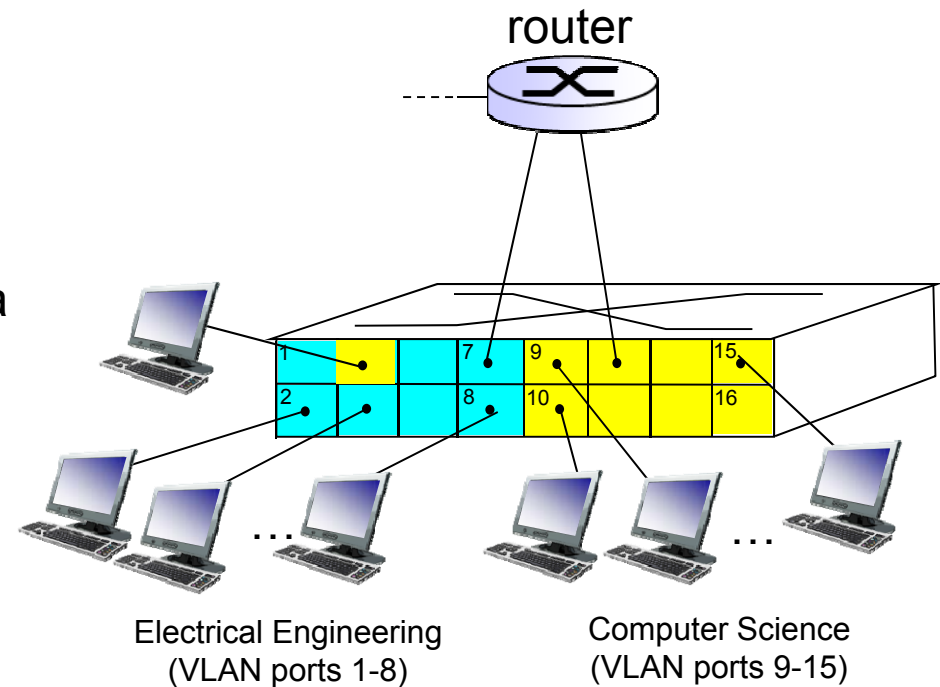


Electrical Engineering  
(VLAN ports 1-8)

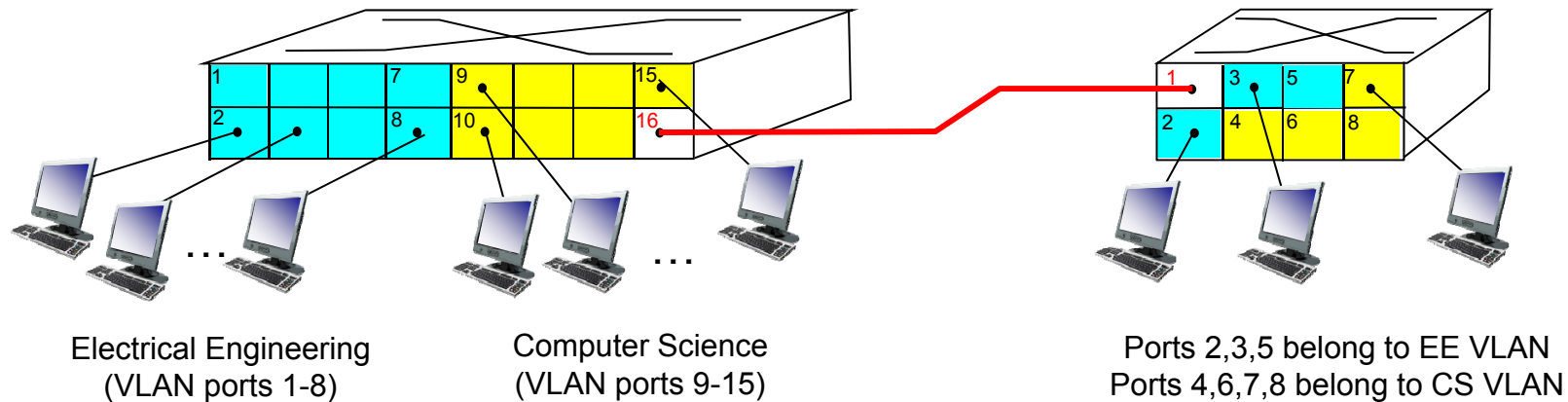
Computer Science  
(VLAN ports 9-16)

# Port-based VLAN

- ❑ *Cô lập lưu lượng:* các frame tới/từ các port 1-8 chỉ có thể tới các port 1-8
  - cũng có thể xác định VLAN dựa trên địa chỉ MAC của các endpoint, thay vì switch port
- ❖ *Thay đổi động:* các port có thể gán động giữa các VLAN
- ❖ *Chuyển tiếp giữa các VLAN:* thực hiện thông qua dẫn đường (tương tự như các switch riêng lẻ)

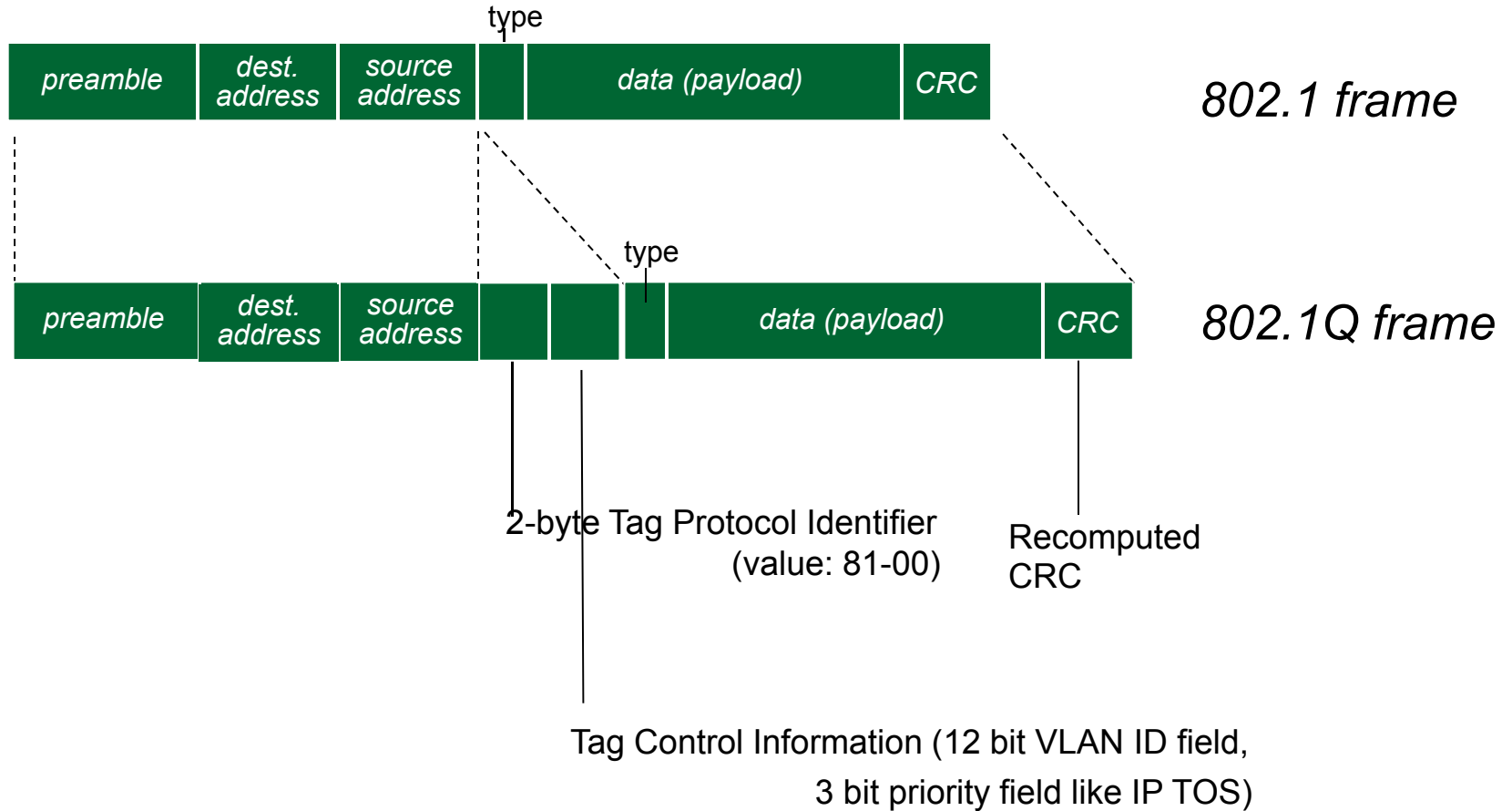


# VLANs spanning multiple switches



- **trunk port:** mang các frame giữa các VLAN được xác định bởi nhiều switch vật lý
  - các frame được chuyển trong VLAN giữa các switch không thể là 802.1 frame (phải có thông tin VLAN ID)
  - giao thức 802.1q thêm/loại bỏ trường header thêm vào cho các frame được chuyển giữa các trunk port

# Cấu trúc 802.1Q VLAN frame



# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

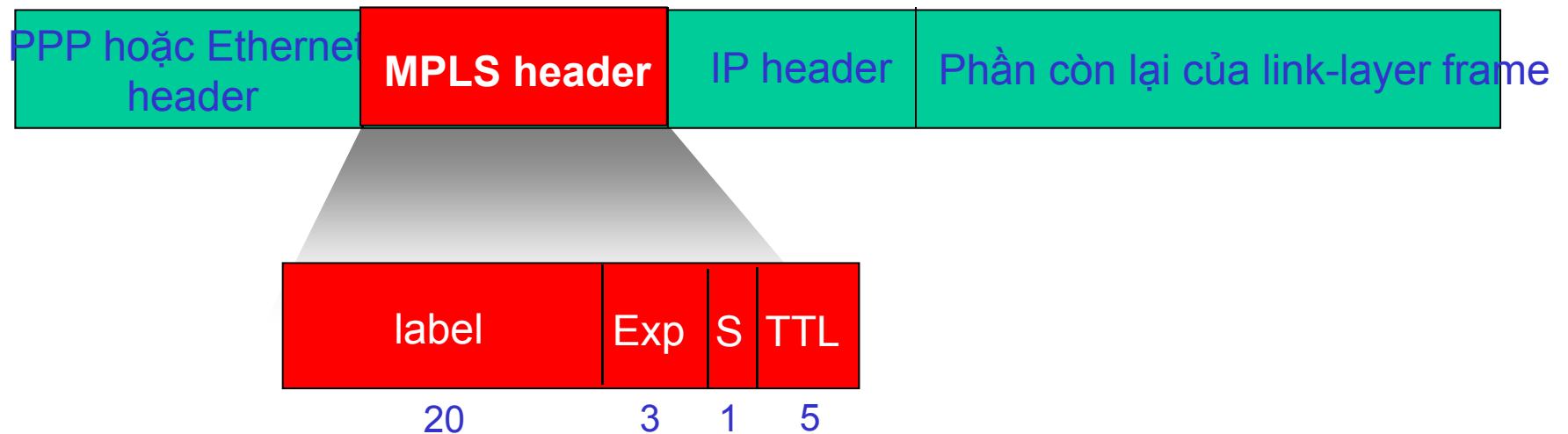
5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu

5.7 Ví dụ liên hệ các kiến thức

# Multiprotocol label switching (MPLS)

- ❑ Mục đích ban đầu: tăng tốc chuyển tiếp IP bằng cách sử dụng nhãn có độ dài cố định (thay vì sử dụng địa chỉ IP) để chuyển tiếp
  - Lấy ý tưởng từ cách tiếp cận Virtual Circuit (VC)
  - Nhưng IP datagram vẫn giữ địa chỉ IP!

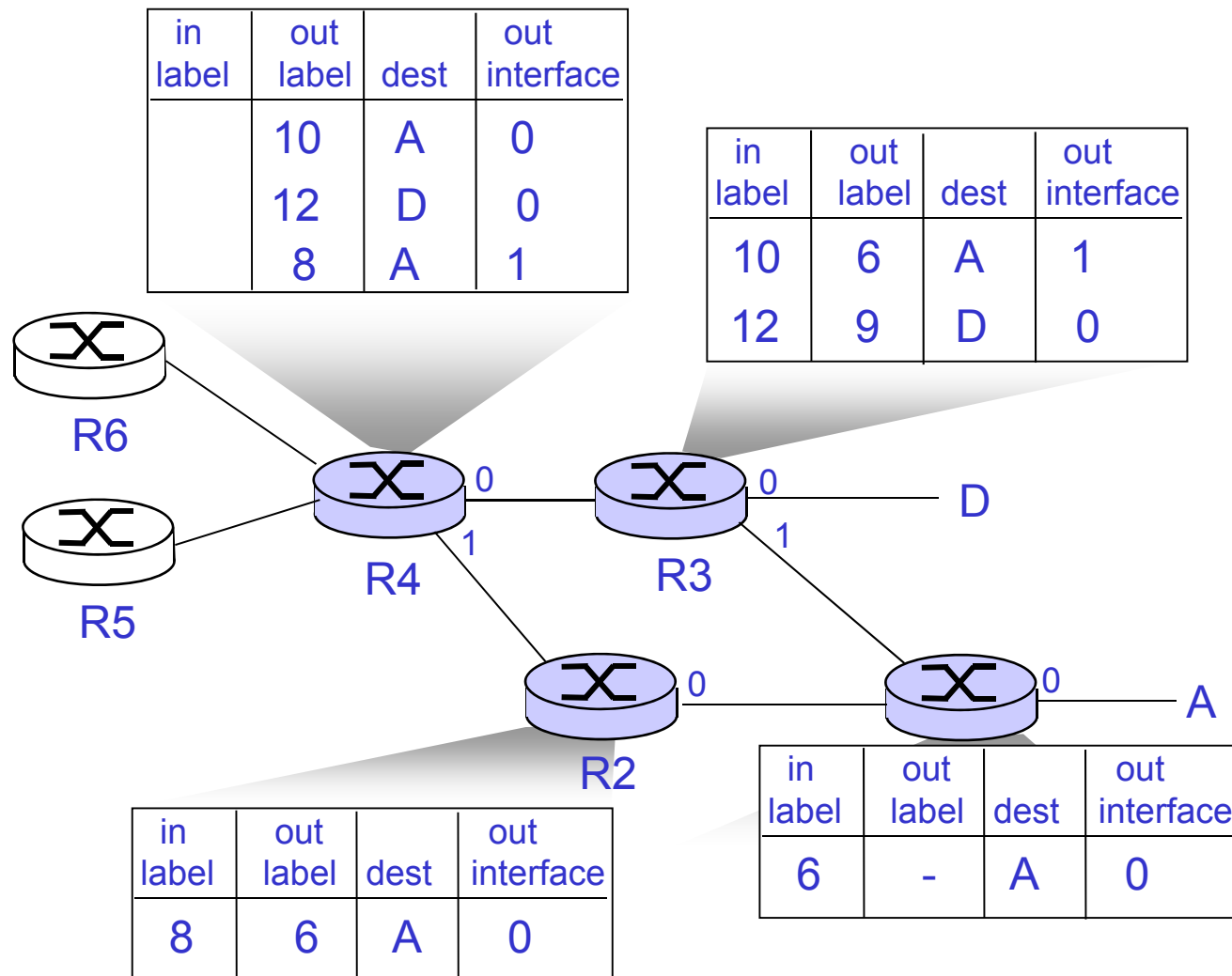


# Các router hỗ trợ MPLS

- ❑ Còn gọi là label-switched router
- ❑ Chuyển tiếp các gói tin tới outgoing interface chỉ dựa trên giá trị nhãn (không kiểm tra địa chỉ IP)
  - Bảng chuyển tiếp MPLS khác bảng chuyển tiếp IP
- ❑ Giao thức bao hiệu cần được thiết lập chuyển tiếp
  - RSVP-TE
  - Chuyển tiếp có thể dọc theo đường mà IP không cho phép (ví dụ: source-specific routing) !!
  - Sử dụng MPLS cho traffic engineering
- ❑ Phải cùng tồn tại với router chỉ hỗ trợ IP



# Bảng chuyển tiếp của MPLS



# Chương 5: Tầng liên kết

5.1 Giới thiệu

5.2 Phát hiện lỗi, sửa lỗi

5.3 Các giao thức đa truy cập

5.4 LAN

- Địa chỉ, ARP
- Ethernet
- Switch
- VLAN

5.5 Ảo hóa liên kết: MPLS

5.6 Mạng trung tâm dữ liệu

# Mạng trung tâm dữ liệu (Data center network)

- ❑ Hàng chục tới hàng trăm nghìn host, xếp gần nhau:
  - e-business (ví dụ: Amazon)
  - content-server (ví dụ: YouTube, Akamai, Apple, Microsoft)
  - search engines, data mining (ví dụ: Google)
  
- ❑ Thách thức:
  - nhiều ứng dụng, mỗi ứng dụng phục vụ số lượng lớn client
  - quản lý/cân bằng tải, tránh điểm nghẽn về xử lý hoặc truyền thông trong mạng

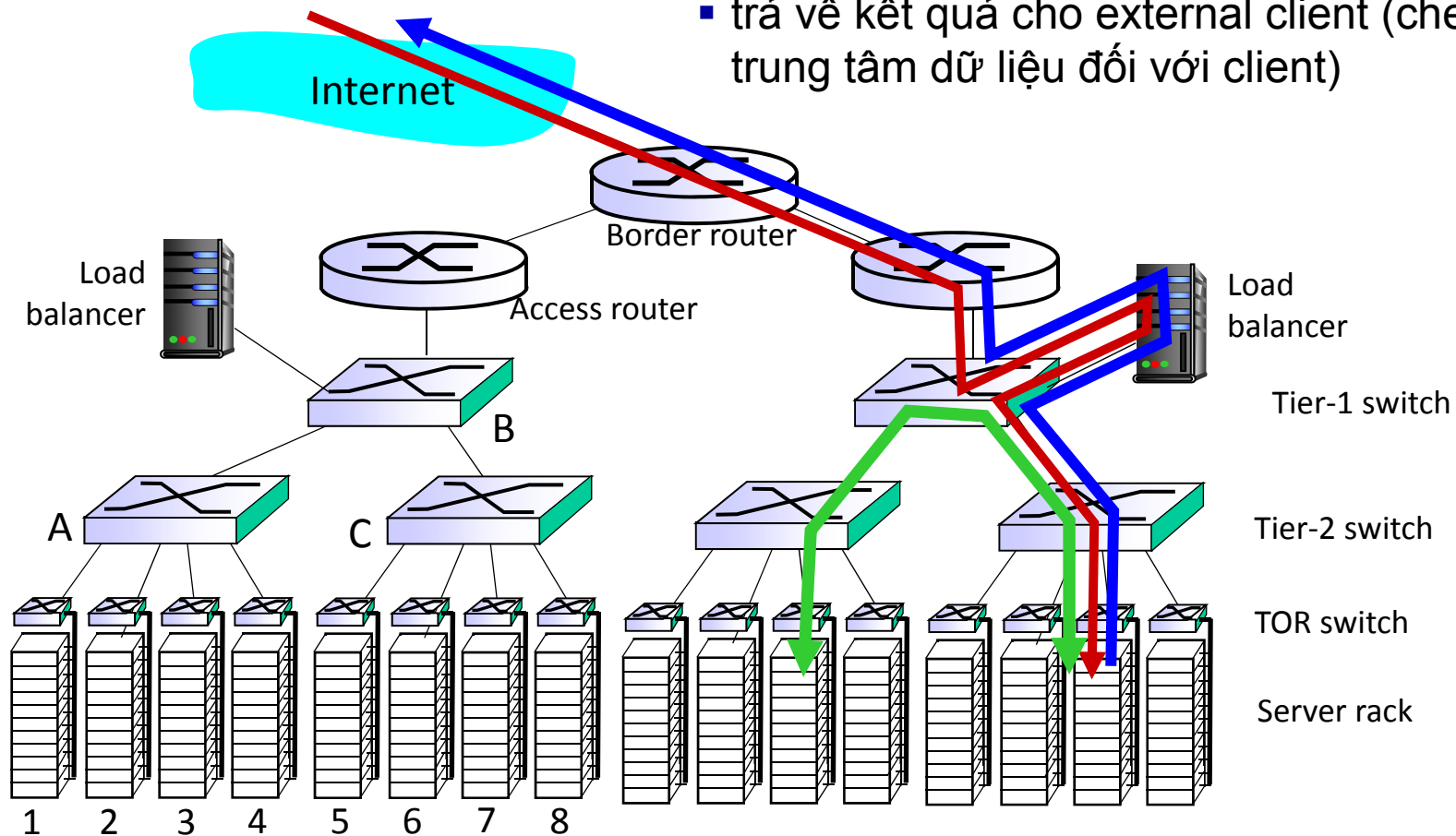


Microsoft: Chicago data center

# Mạng trung tâm dữ liệu

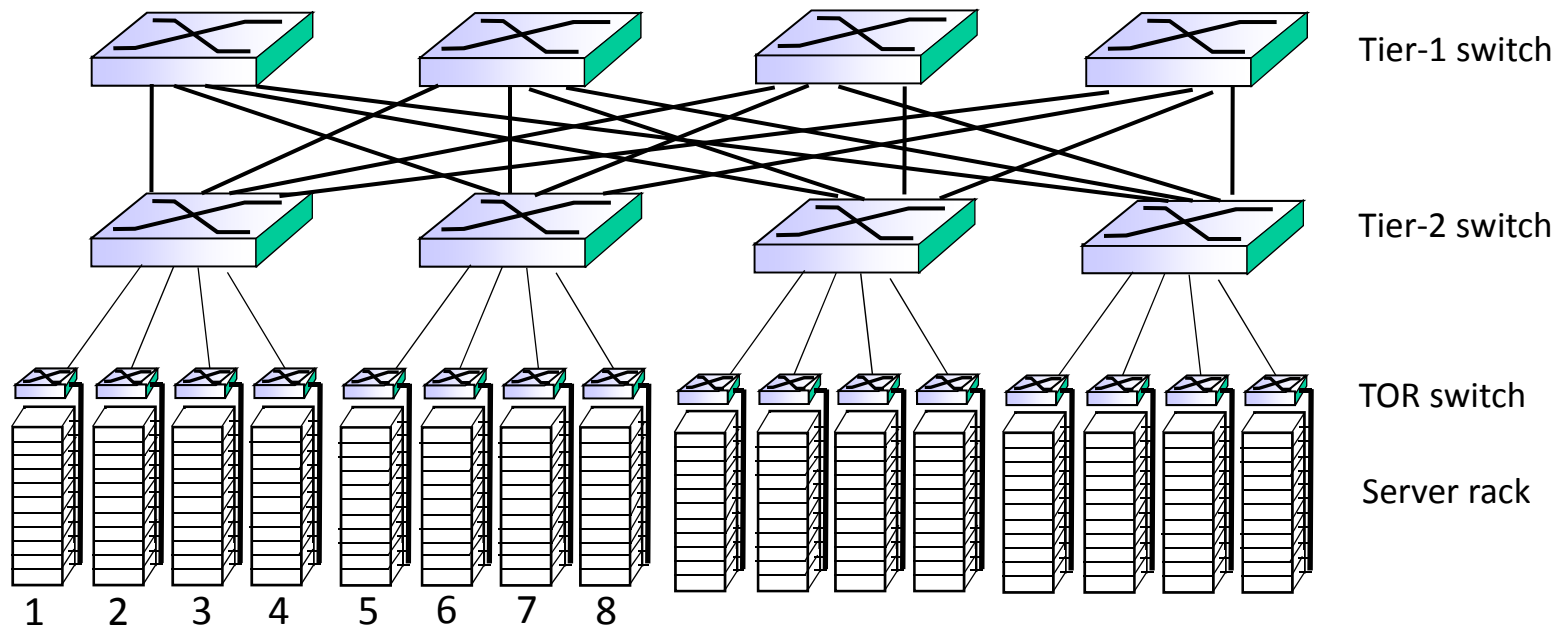
*Cân bằng tải: dẫn đường ở tầng ứng dụng*

- nhận các yêu cầu của external client
- chuyển workload trong data center
- trả về kết quả cho external client (che dấu trung tâm dữ liệu đối với client)



# Mạng trung tâm dữ liệu

- Kết nối dây giữa các switch, rack:
  - thông lượng tăng giữa các rack (có thể dùng nhiều đường trong định tuyến)
  - tăng độ tin cậy bằng sự dư thừa



# Chương 5: Tổng kết

- ❑ Các nguyên tắc bên trong của các dịch vụ tầng liên kết dữ liệu:
  - Phát hiện và sửa lỗi
  - Dùng chung một kênh quảng bá: đa truy cập
  - Đánh địa chỉ tầng liên kết dữ liệu
- ❑ Ví dụ và thực hiện của các công nghệ tầng liên kết dữ liệu khác nhau
  - Ethernet
  - Switched LANS
  - Virtualized network từ tầng liên kết dữ liệu: ATM, MPLS

# Bài tập thực hành

- ❑ Sử dụng Wireshark để quan sát hoạt động của giao thức IEEE 802.3

# An interview with Simon S. Lam

- ❑ Simon S. Lam is Professor and Regents Chair in Computer Sciences at the University of Texas at Austin. He led a research group that invented secure sockets and prototyped, in 1993, the first secure sockets layer named Secure Network Programming, which won the 2004 ACM Software System Award. He received his BSEE from Washington State University and his MS and PhD from UCLA. He was elected to the National Academy of Engineering in 2007.
- ❑ What impacts do you foresee technology having on learning in the future?
  - Eventually, almost all human knowledge will be accessible through the Internet, which will be the most powerful tool for learning. This vast knowledge base will have the potential of leveling the playing field for students all over the world. For example, motivated students in any country will be able to access the best-class Web sites, multimedia lectures, and teaching materials. Already, it was said that the IEEE and ACM digital libraries have accelerated the development of computer science researchers in China. In time, the Internet will transcend all geographic barriers to learning.



# Mạng máy tính

- Hình ảnh và nội dung trong bài giảng này có tham khảo từ sách và bài giảng của TS. J.F. Kurose and GS. K.W. Ross