

Università degli Studi di Roma "Tor Vergata"  
Laurea in Informatica

Sistemi Operativi e Reti  
(modulo Reti)  
a.a. 2024/2025

# Livello di collegamento (parte3)

dr. Manuel Fiorelli

[manuel.fiorelli@uniroma2.it](mailto:manuel.fiorelli@uniroma2.it)

<https://art.uniroma2.it/fiorelli>

Basate sulle slide del libro di testo:

[https://gaia.cs.umass.edu/kurose\\_ross/ppt.php](https://gaia.cs.umass.edu/kurose_ross/ppt.php)

# Livello di collegamento e LAN: tabella di marcia

- introduzione
- rilevazione e correzione degli errori
- protocolli di accesso multiplo
- **LAN**
  - indirizzamento, ARP
  - Ethernet
  - switch
  - VLAN
- canali virtuali: MPLS
- Reti dei data center



- un giorno nella vita di una richiesta web

# LAN

## Local Area Network (LAN)

Copre un'area limitata come un'abitazione, una scuola, un ufficio o un edificio (o gruppo di edifici vicini).

Due tecnologie principali:

- Ethernet (questa tecnologia è usata anche in altri ambiti): IEEE 802.3 (nome del working group dell'IEEE e della famiglia di standard)
- Wi-Fi: IEEE 802.11

# Indirizzi MAC

- indirizzi IP a 32 bit (128 bit in IPv6):
  - indirizzi *a livello di rete* per le interfacce
  - usati per l'inoltro a livello 3 (livello di rete)
  - es.: 128.119.40.136
- Indirizzi MAC (o LAN o fisici o Ethernet):
  - funzione: utilizzati “localmente” per portare i frame da un'interfaccia a un'altra interfaccia fisicamente connessa (stessa sottorete, nel senso dell'indirizzamento IP)
  - indirizzo MAC a 48 bit (per la maggior parte delle LAN) memorizzato nella ROM della NIC, a volte impostabile via software.
  - es.: 1A-2F-BB-76-09-AD

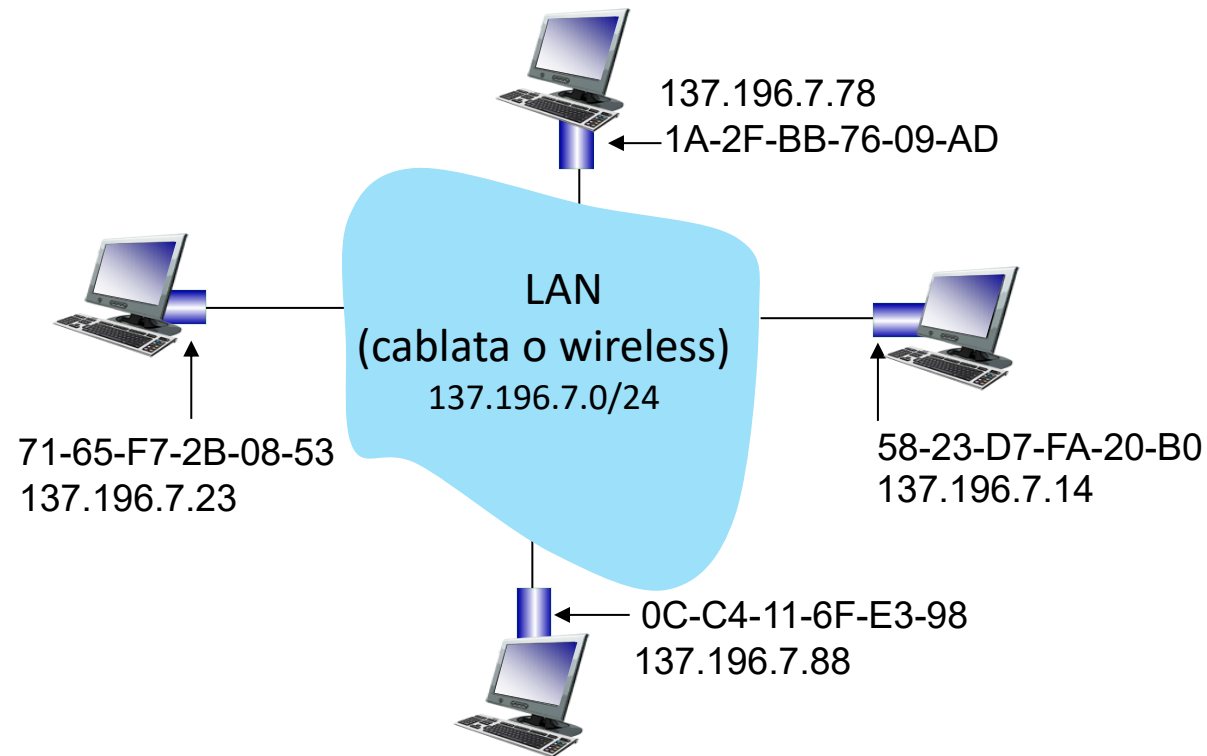
2001:0df0:00f2:0000:0000:0000:0000:0f10 → 2001:df0:f2::f10

notazione esadecimale (base 16)  
(ciascuna “cifra” rappresenta 4 bit)

# Indirizzi MAC

ciascuna interfaccia in una LAN

- ha un indirizzo **MAC** univoco
- ha un indirizzo IP univoco (come abbiamo visto)

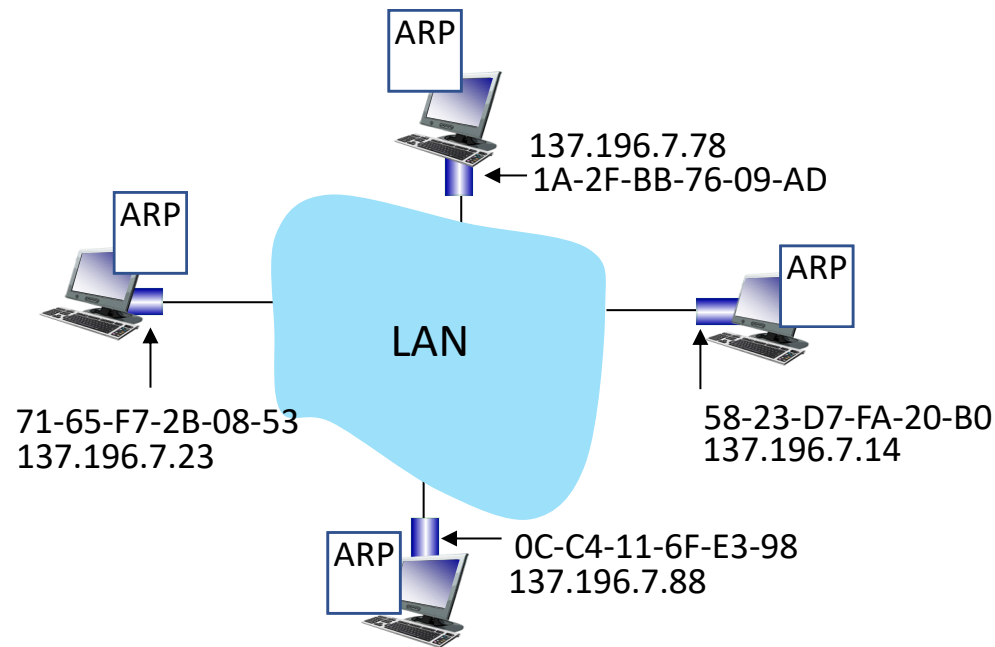


# Indirizzi MAC

- allocazione degli indirizzi MAC gestita dall'IEEE
- i produttori (di schede di rete) comprano porzioni dello spazio di indirizzi MAC (per assicurare l'unicità)
- analogia:
  - indirizzi MAC: come il codice fiscale
  - indirizzo IP: come l'indirizzo postale
- indirizzo MAC (piatto): portabilità
  - è possibile spostare un'interfaccia da una LAN a un'altra
  - indirizzo IP (gerarchico) *non* portabile: dipende dalla sottorete IP alla quale il nodo è connesso

# Protocollo per la risoluzione degli indirizzi (*address resolution protocol, ARP*)

**Domanda:** come determinare l'indirizzo MAC di un'interfaccia, conoscendo il suo indirizzo IP?



**Tabella ARP:** ogni nodo IP (host, router) sulla LAN ha una tabella (una per ciascuna interfaccia)

- corrispondenza tra indirizzi IP e MAC per alcuni nodi sulla LAN:  
< indirizzo IP; indirizzo MAC address; TTL >
- TTL (Time To Live): tempo dopo il quale la mappatura degli indirizzi sarà dimenticata (in genere 20 min da quando la voce è stata inserita nella tabella)

# Protocollo ARP in azione

esempio: A vuole inviare un datagramma a B

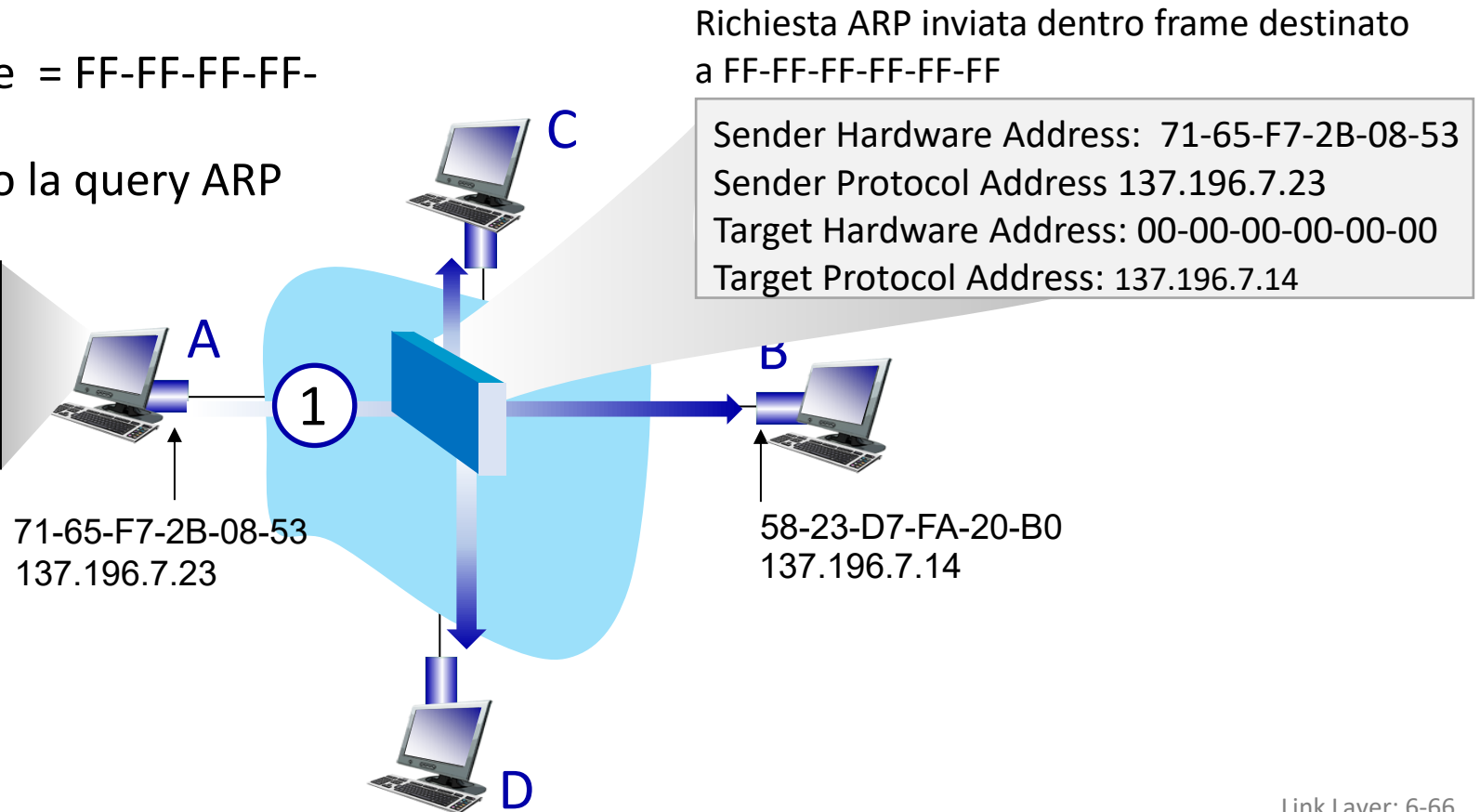
- l'indirizzo MAC di B non è nella tabella ARP di A, pertanto A usa ARP per trovare l'indirizzo MAC di B

A invia in broadcast una richiesta ARP,

- ① contenente l'indirizzo IP di B
- indirizzo MAC di destinazione = FF-FF-FF-FF-FF-FF
  - tutti i nodi sulla LAN ricevono la query ARP

Tabella ARP in A

IP addr	MAC addr	TTL





# Protocollo ARP in azione

esempio: A vuole inviare un datagramma a B

- l'indirizzo MAC di B non è nella tabella ARP di A, pertanto A usa ARP per trovare l'indirizzo MAC di B

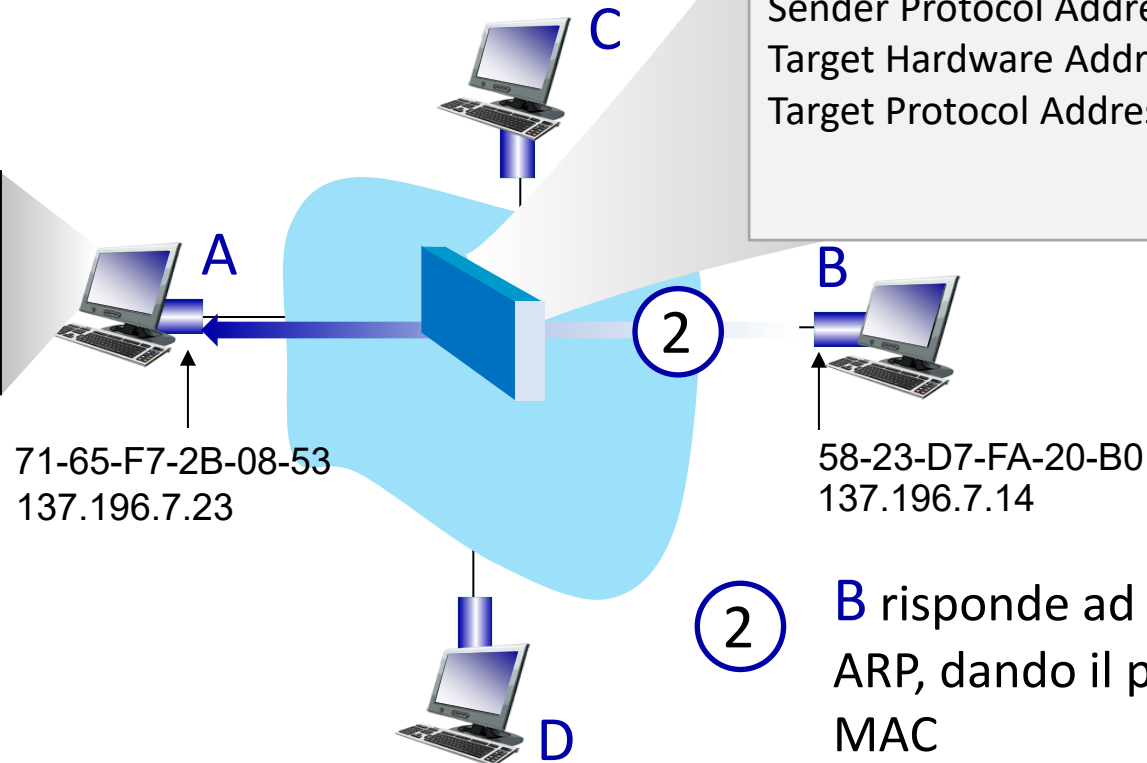
source IP/MAC e target IP/MAC  
nella risposta sono invertiti rispetto  
alla richiesta

Risposta ARP inviata dentro frame  
destinato a 71-65-F7-2B-08-53

Sender Hardware Address: 58-23-D7-FA-20-B0  
Sender Protocol Address: 137.196.7.14  
Target Hardware Address: 71-65-F7-2B-08-53  
Target Protocol Address 137.196.7.23

Tabella ARP in A

IP addr	MAC addr	TTL

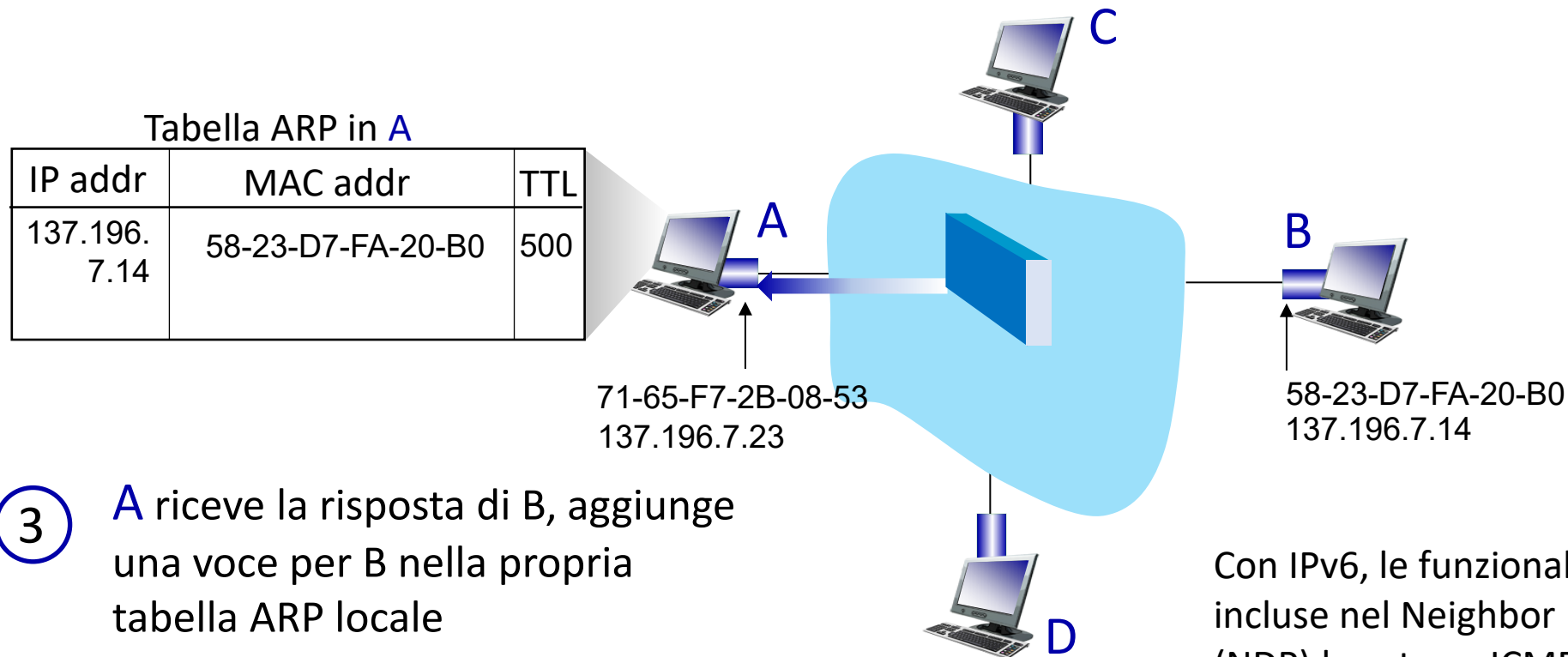


B risponde ad A con una risposta  
ARP, dando il proprio indirizzo  
MAC

# Protocollo ARP in azione

esempio: A vuole inviare un datagramma a B

- l'indirizzo MAC di B non è nella tabella ARP di A, pertanto A usa ARP per trovare l'indirizzo MAC di B



Con IPv6, le funzionalità di ARP sono state incluse nel Neighbor Discovery Protocol (NDP) basato su ICMPv6.

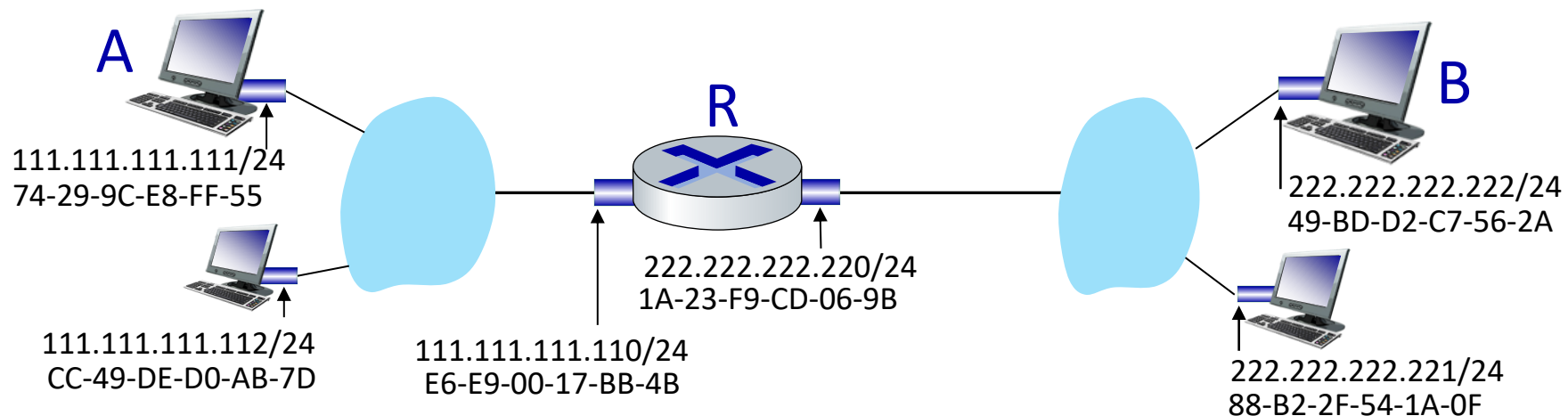
# ARP Spoofing o ARP Poisoning

- Un attaccante invia in una LAN risposte ARP contraffatte, inducendo l'associazione di un indirizzo IP a un certo indirizzo MAC
- Il protocollo ARP è senza stato e un nodo (host o router) aggiorna la propria ARP appena viene ricevuta una risposta ARP (a prescindere che questa faccia seguito a una effettiva richiesta)
- Alcuni "usi":
  - *denial-of-service* (DoS): associare diversi indirizzi IP allo stesso indirizzo MAC per sovraccaricarlo di traffico
  - *man-in-the-middle* (MITM): l'attaccante associa il proprio indirizzo MAC all'indirizzo IP di un altro nodo, in modo da intercettare (e magari modificare) il traffico destinato a quest'ultimo, per poi re-inoltrarglielo

# Come inviare un datagramma a un nodo esterno alla sottorete

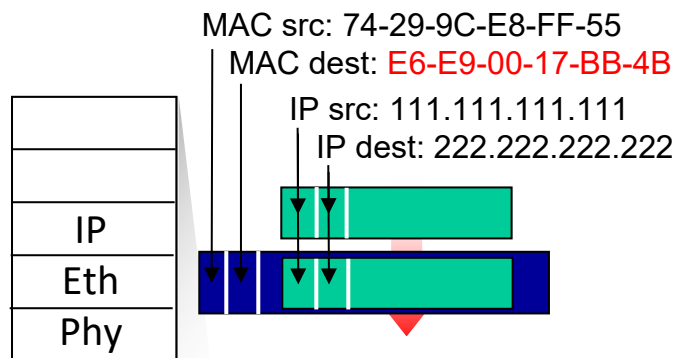
scenario dettagliato: **invio di un datagramma da A a B passando per R**

- attenzione sugli indirizzi – a livello IP (datagramma) e MAC (frame)
- assunzioni:
  - A conosce l'indirizzo IP di B
  - A conosce l'indirizzo IP dell'interfaccia di R nella propria sottorete (come? DHCP)
  - A conosce l'indirizzo MAC dell'interfaccia di R nella propria sottorete (come? ARP)

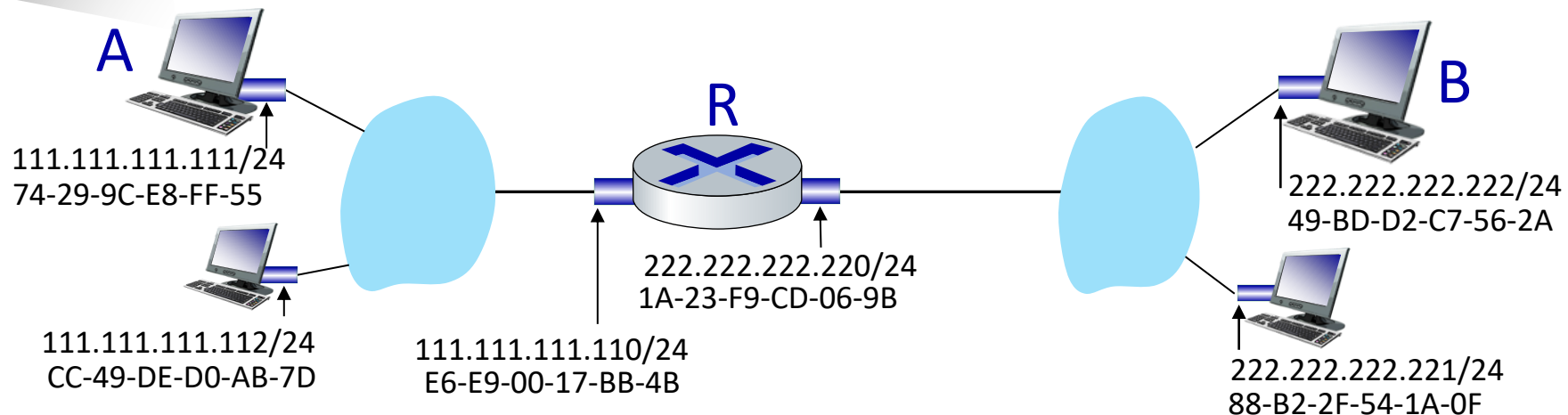


# Come inviare un datagramma a un nodo esterno alla sottorete

- A crea un datagramma IP con sorgente A e destinazione B
- A crea un frame a livello di collegamento contenente il datagramma IP da A a B
  - la destinazione del frame è l'indirizzo MAC di **R**

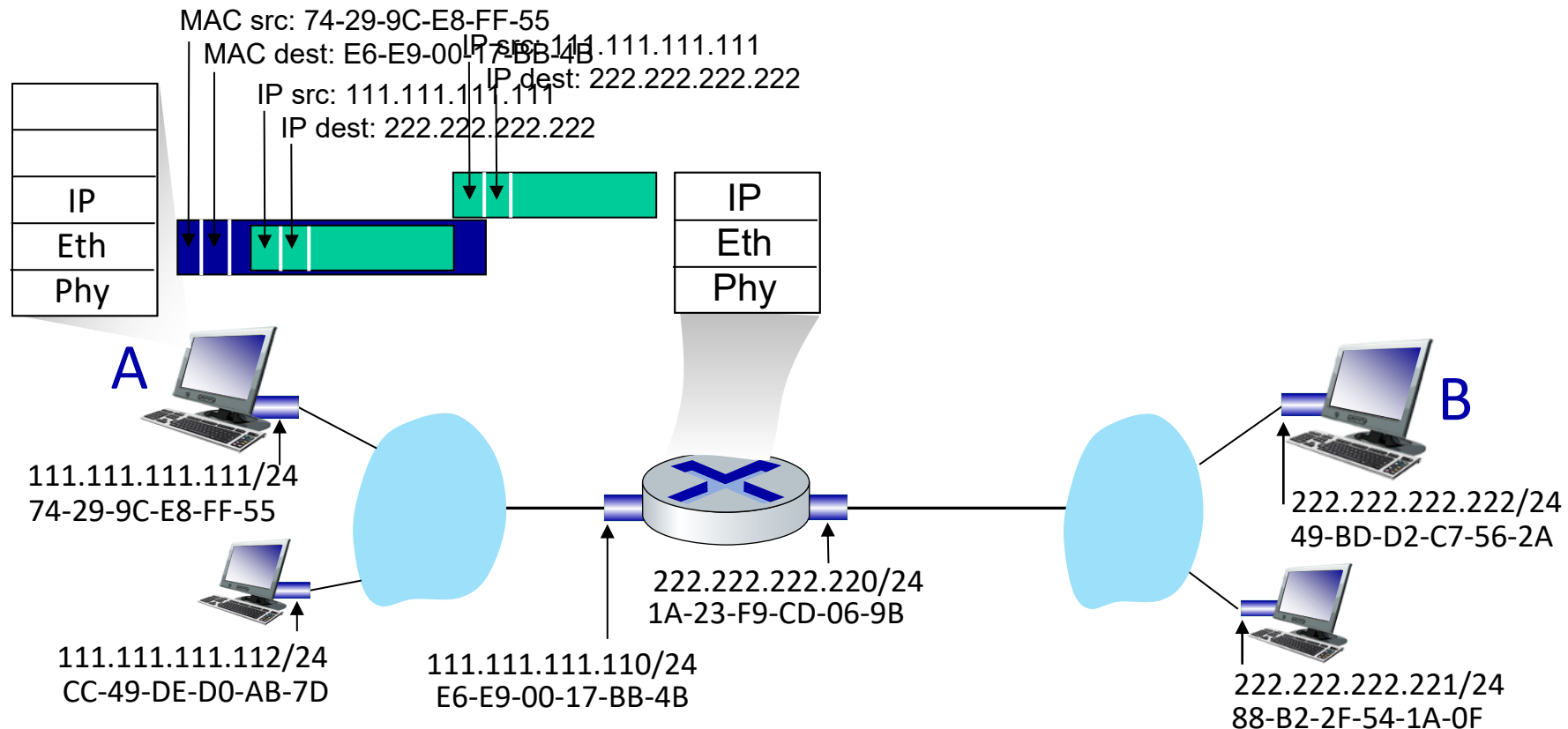


A sa di appartenere a una sottorete /24, pertanto confronta i 24 bit più significativi del proprio indirizzo con quelli dell'indirizzo di B, constatando che sono diversi e che quindi B si trova in una sottorete differente



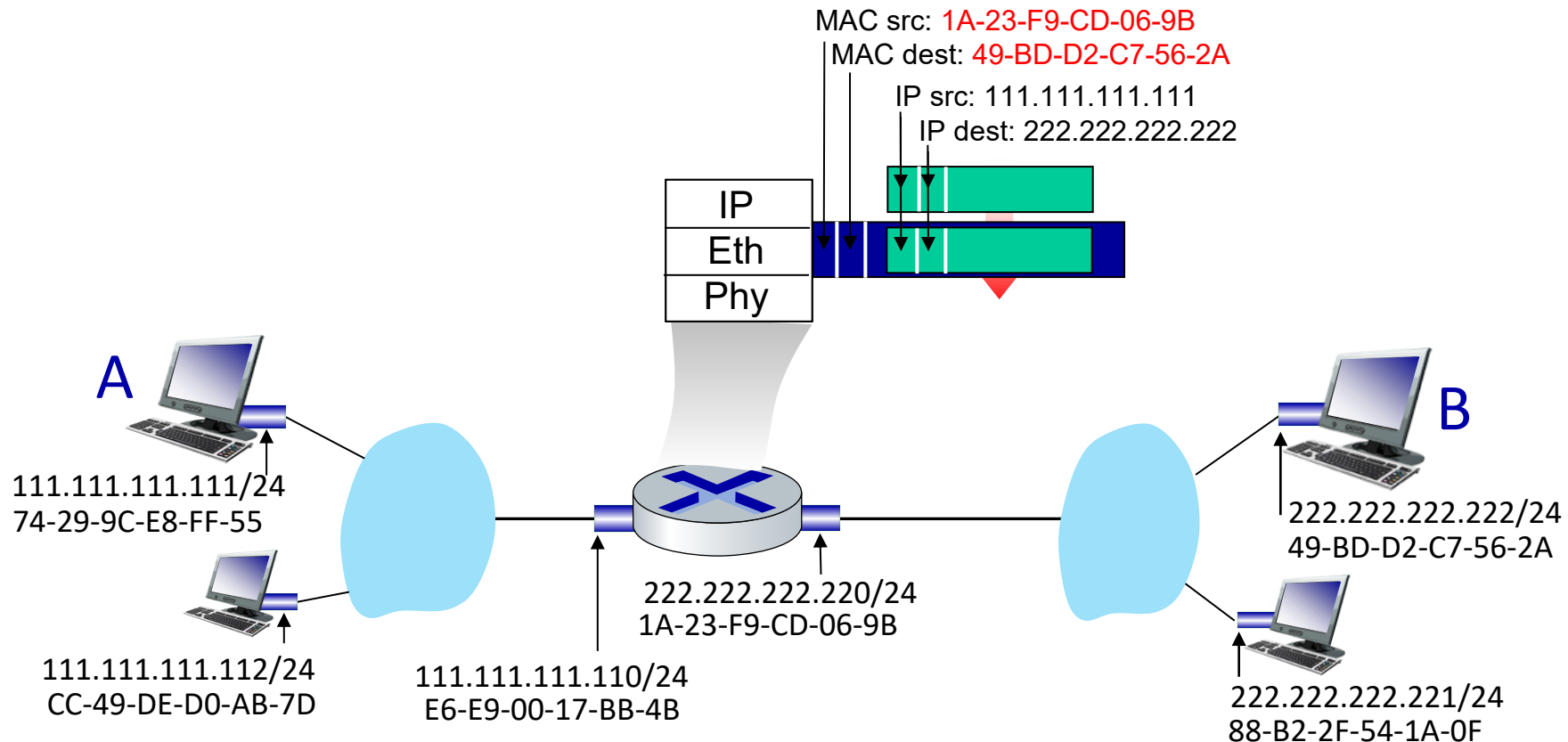
# Come inviare un datagramma a un nodo esterno alla sottorete

- frame inviato da A a R
- frame ricevuto da R, datagramma, passato in alto a IP



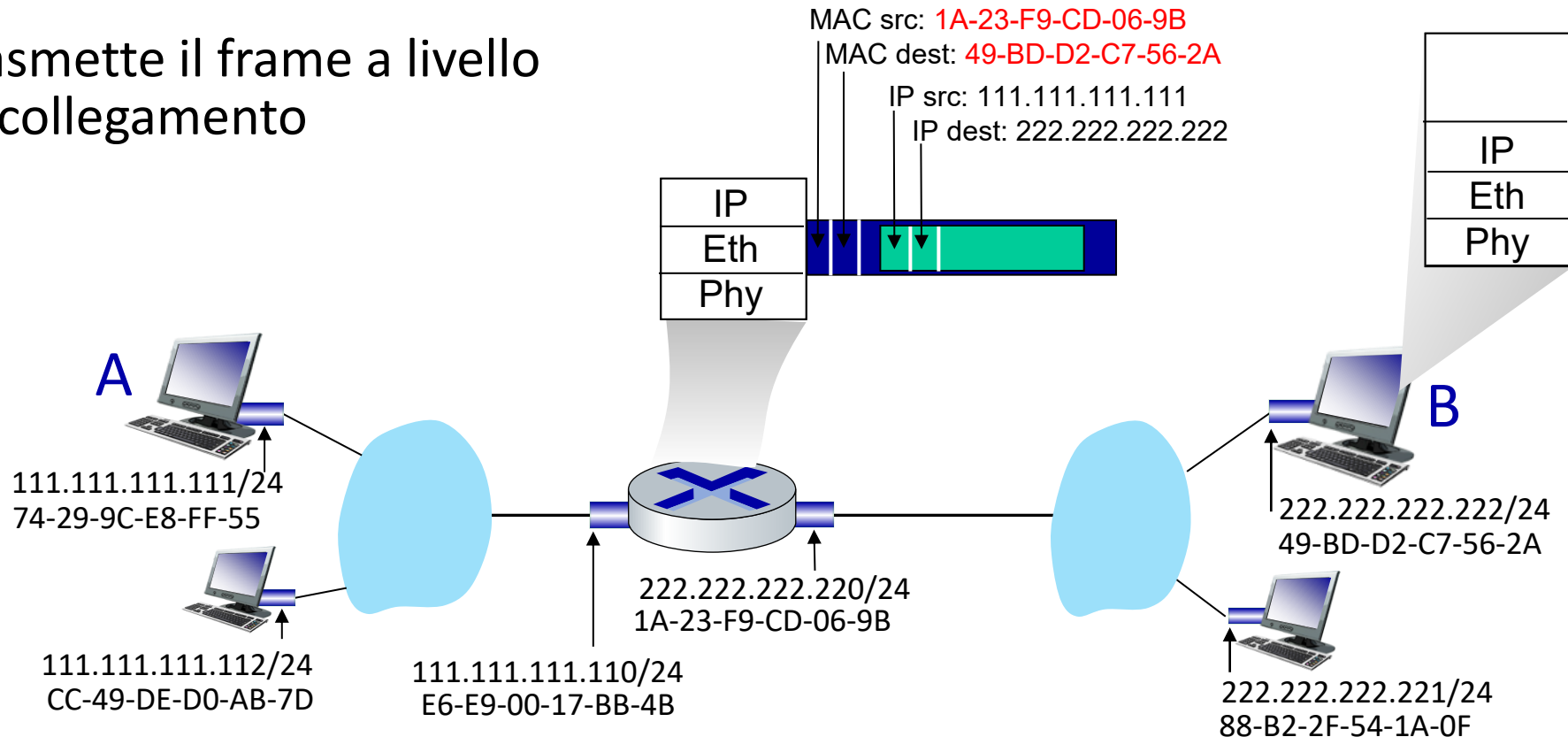
# Come inviare un datagramma a un nodo esterno alla sottorete

- R determina l'interfaccia di uscita, passa il datagramma con sorgente IP A e destinazione IP B al livello di collegamento
- R crea il frame a livello di collegamento contenente il datagramma IP da A a B. Indirizzo di destinazione del frame: indirizzo MAC di B



# Come inviare un datagramma a un nodo esterno alla sottorete

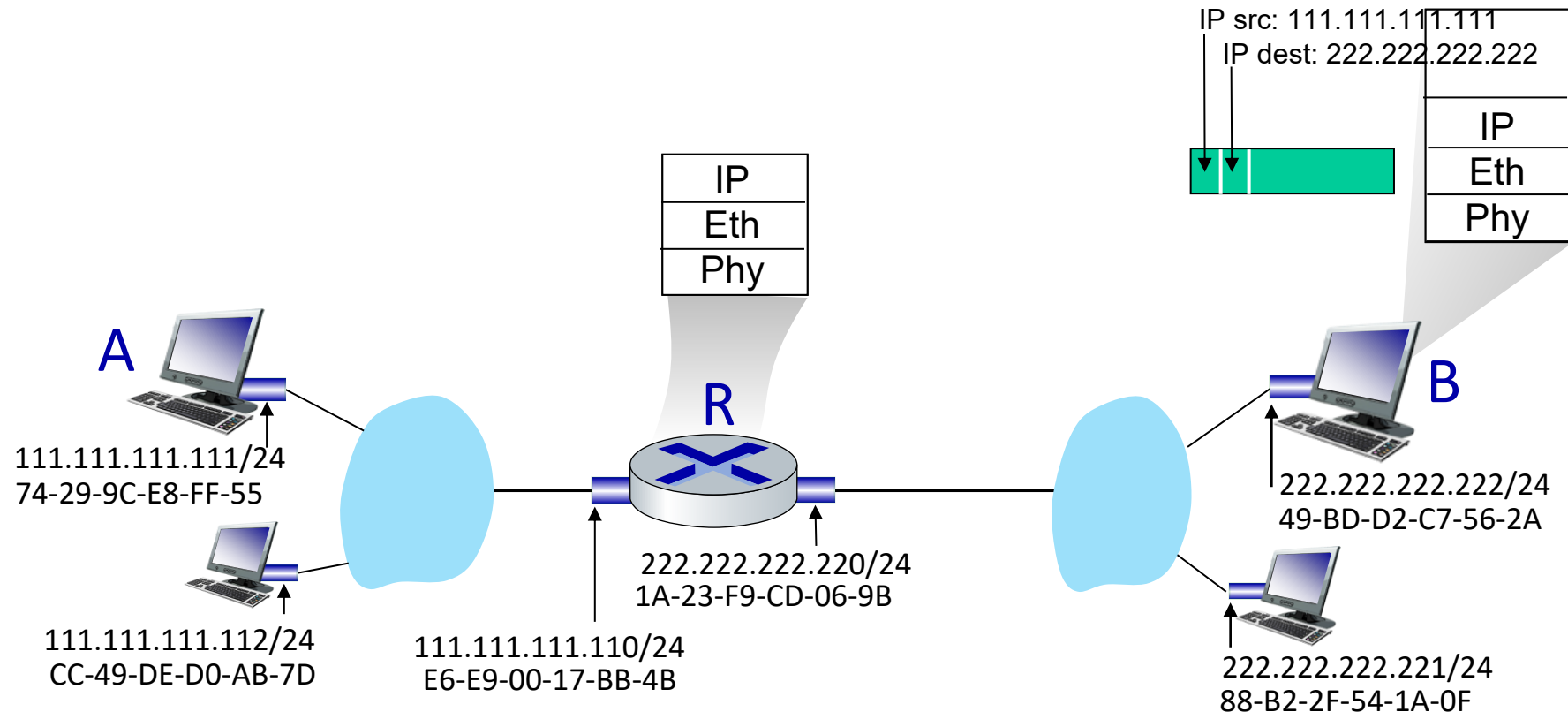
- R determina l'interfaccia di uscita, passa il datagramma con sorgente IP A e destinazione IP B al livello di collegamento
- R crea il frame a livello di collegamento contenente il datagramma IP da A a B. Indirizzo di destinazione del frame: indirizzo MAC di B
- trasmette il frame a livello di collegamento





# Come inviare un datagramma a un nodo esterno alla sottorete

- B riceve il frame, il datagramma IP destinato a sé
- B passa il datagramma in alto nella pila protocollare a IP



# Livello di collegamento e LAN: tabella di marcia

- introduzione
- rilevazione e correzione degli errori
- protocolli di accesso multiplo
- **LAN**
  - indirizzamento, ARP
  - **Ethernet**
  - switch
  - VLAN
- canali virtuali: MPLS
- Reti dei data center



- un giorno nella vita di una richiesta web

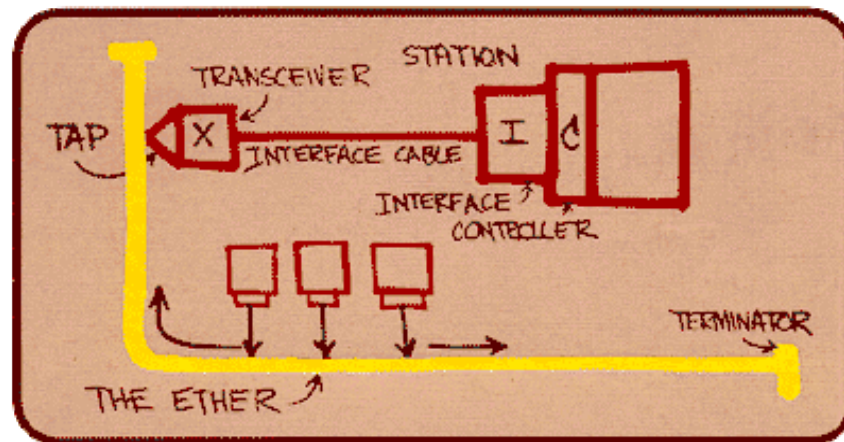
# Ethernet

Tecnologia “dominante” per le LAN cablate:

- prima tecnologia LAN ampiamente utilizzata
- semplice, economica
- ha tenuto il passo sulla velocità: 10 Mbps – 400 Gbps
- singolo chip, più velocità (es., Broadcom BCM5761)

Bob Metcalfe: co-inventore di Ethernet,  
destinatario del Premio ACM Turing 2022

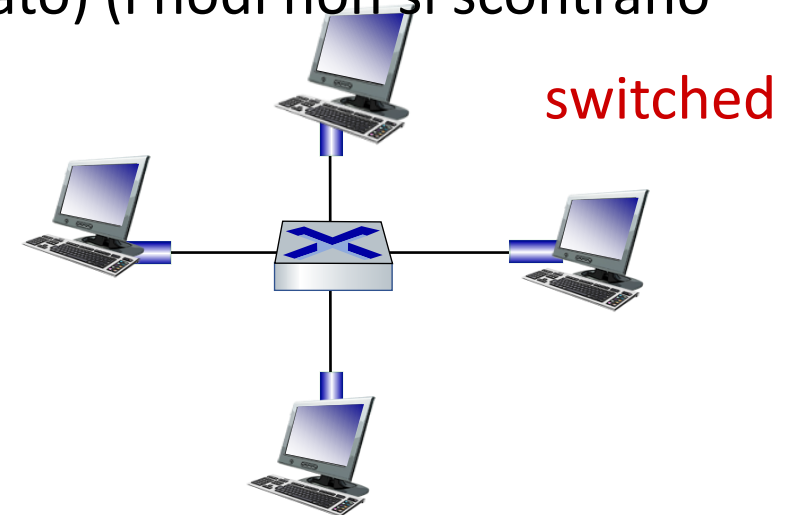
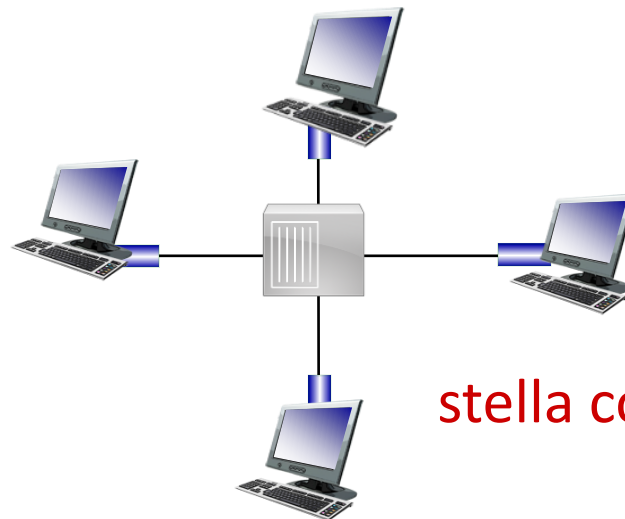
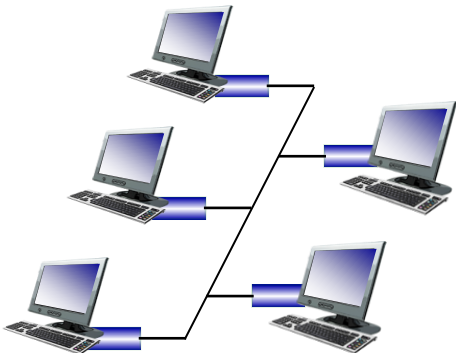
*Schizzo Ethernet di  
Metcalfe*



# Ethernet: topologia fisica

- **bus:** popolare fino alla metà degli anni '90
  - tutti i nodi sono nello stesso dominio di collisione (possono collidere tra loro)
- **topologia a stella con hub:** popolare fino agli anni 2000
  - i nodi sono interconnessi da un hub (dispositivo a livello fisico che rigenera i segnali ricevuti su una interfaccia e li ritrasmette su tutte le altre interfacce), pertanto tutti i nodi sono nello stesso dominio di collisione
- **commutata (switched):** oggi prevalente
  - *switch* di livello 2 attivo al centro
  - ogni “spoke” esegue un protocollo Ethernet (separato) (i nodi non si scontrano tra loro)

**bus:** cavo coassiale



# Struttura del frame Ethernet

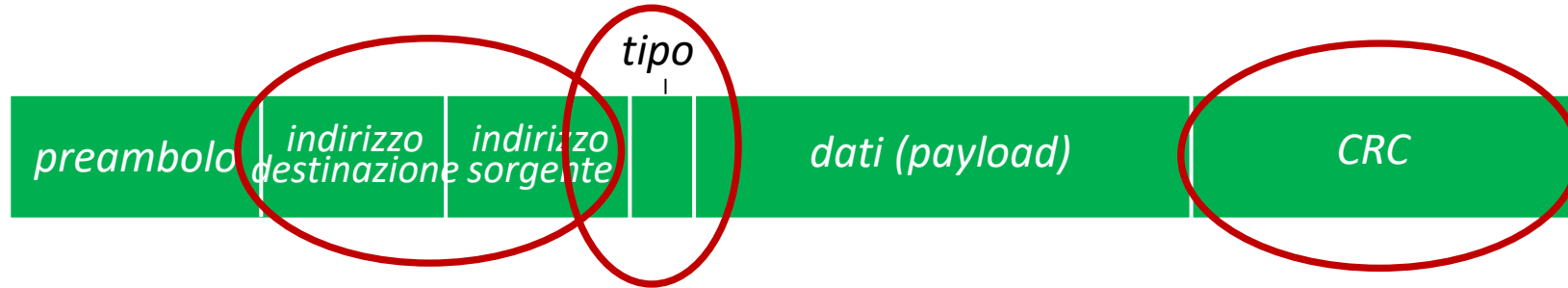
l'interfaccia trasmittente incapsula il datagramma IP (o altro pacchetto di protocolli di livello di rete) in **frame Ethernet**



## *preambolo:*

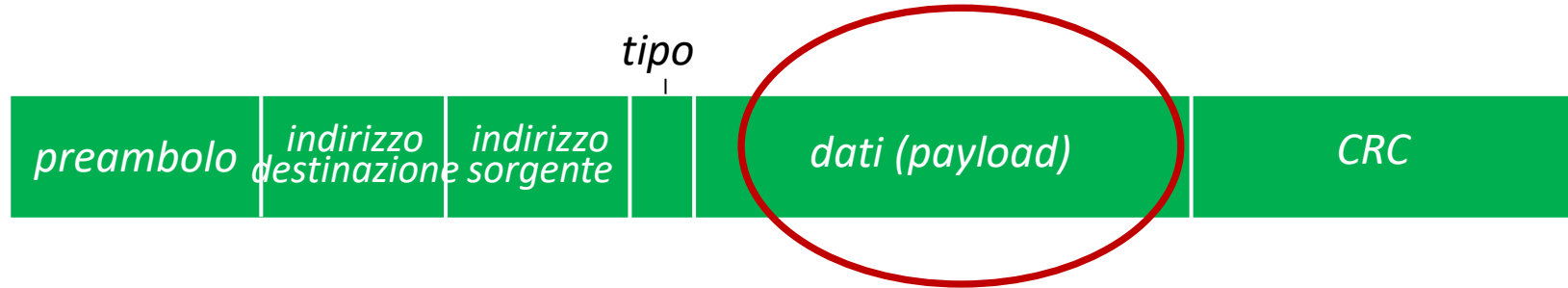
- usato per “risvegliare” le schede di rete dei riceventi e sincronizzare i loro clock con quello del trasmittente
  - 7 byte di 10101010 seguiti da un byte di 10101011
- questi due 1 consecutivi, che rompono il pattern di 1 e 0 alternati, informano il ricevente dell'inizio del frame vero e proprio

# Struttura del frame Ethernet (continuazione)



- **indirizzi:** indirizzi sorgente e destinazione a 6 byte
  - se l'adattatore riceve un frame con un indirizzo di destinazione corrispondente o con un indirizzo di broadcast (ad esempio, un pacchetto ARP), passa i dati nel frame al protocollo di livello superiore
  - altrimenti, l'adattatore scarta il frame
- **tipo:** indica un protocollo di livello superiore (2 byte)
  - principalmente IP, ma sono possibili anche altri protocolli a livello di rete, ad es. Novell IPX, AppleTalk, ma anche ARP
  - utilizzato per demultiplexare sul ricevitore
- **CRC:** controllo di ridondanza ciclica presso il ricevitore (4 byte)
  - errore rilevato: il frame viene scartato

# Struttura del frame Ethernet (continuazione)



- **payload:** dati passati al protocollo di livello superiore
  - minimo 46 byte: se più piccolo deve essere aggiunto del padding; perciò è importante che il protocollo di livello superiore preveda un meccanismo per stabilire la reale dimensione dei dati
  - massimo 1500 byte (salvo estensioni) → MTU
- La **fine del frame** è determinata a livello fisico dall'assenza di transizioni sulla linea
- escludendo il preambolo, la **dimensione di un frame** è compresa tra 64 byte (512 bit) e 1518 byte → slot time 512 bit. In Gigabit Ethernet lo slot time è in realtà di 4096 bit (512 byte): frame più piccoli richiedono l'aggiunta di padding (fatto in maniera trasparente dall'hardware)

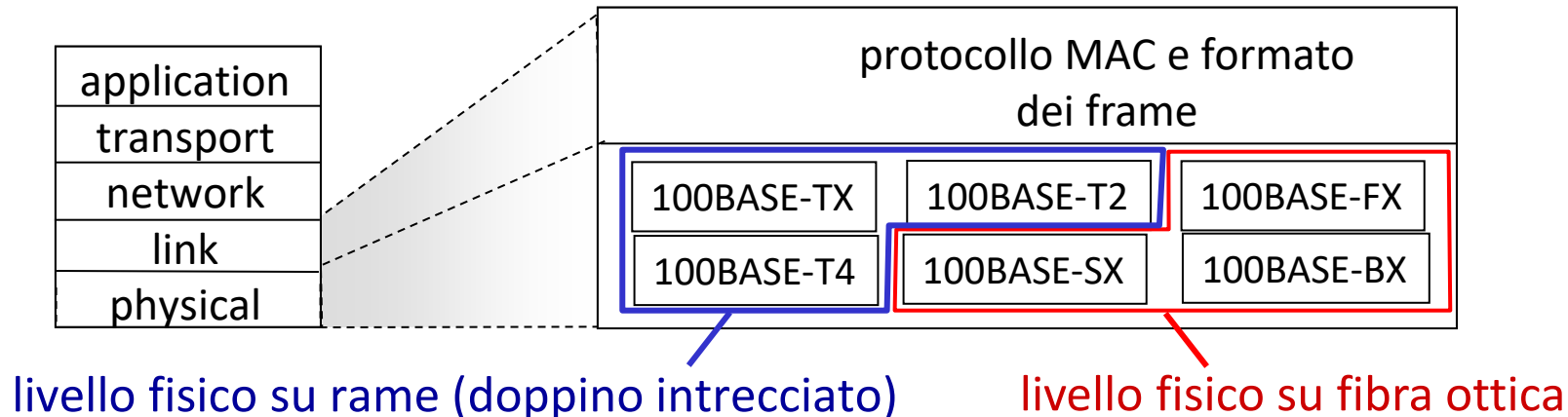
# Ethernet: non affidabile, senza connessione

- **senza connessione:** nessun handshake tra le NIC mittente e ricevente
- **non affidabile:** la NIC ricevente non invia ACK o NAK alla NIC mittente
  - i dati nei frame scartati vengono recuperati solo se il mittente iniziale utilizza un trasferimento dati affidabile di livello superiore (ad esempio, TCP), altrimenti i dati scartati vanno persi
- Protocollo MAC di Ethernet: "unslotted" **CSMA/CD con binary backoff**



# 802.3 Ethernet standard: livelli di collegamento e fisico

- *molti* standard Ethernet differenti
  - protocollo MAC e formato dei frame comuni
  - velocità differenti: 2 Mbps, ... 100 Mbps, 1Gbps, 10 Gbps, 40 Gbps, 100 Gbps
    - mezzi trasmissioni differenti: cavo coassiale, doppino, fibra
  - limiti di lunghezza:
    - segmento: 100 m su doppino intrecciato (Cat. 5) per Fast Ethernet (100 Mbps) e Gigabit Ethernet (1 Gbps); da qualche centinaio di metri fino a decine chilometri in base alla classe di fibra ottica
    - dominio di collisione: il ritardo round-trip deve essere inferiore allo slot time: occorre considerare il ritardo dovuto alla propagazione nel mezzo, così come quello introdotto da hub o repeater, dagli adattatore, etc.: ~2500 m 10BASE-T; ~ 205 m 100BASE-TX; Gigabit Ethernet solitamente usato in modalità full-duplex, no CSMA/CD



# System considerations for multisegment 100BASE-T networks

Table 29-3—Network component delays, Transmission System Model 2

Component	Round trip delay in bit times per meter	Maximum round trip delay in bit times
Two TX/FX DTEs		100
Two T4 DTEs		138
Two T2 DTEs		96
One T2 or T4 and one TX/FX DTE <sup>a</sup>		127
Cat 3 cabling segment	1.14	114 (100 m)
Cat 4 cabling segment	1.14	114 (100 m)
Cat 5 cabling segment	1.112	111.2 (100 m)
STP cabling segment	1.112	111.2 (100 m)
Fiber optic cabling segment	1.0	412 (412 m)
Class I repeater		140
Class II repeater with all ports TX/FX		92
Class II repeater with any port T4		67
Class II repeater with any port T2		90

<sup>a</sup> Worst-case values are used (TX/FX values for MAC transmit start and MDI input to collision detect; T4 value for MDI input to MDI output).

2 cavi da 100 m, uno per ciascun host

margine di sicurezza

$$127 + 2 \cdot 111.2 + 140 + 4 = 493,4 < 512$$

Considero una rete Fast Ethernet con topologia a stella (con hub)

I ritardi sono espresso in *bit time*, ovvero il tempo impiegato per immettere un bit nel collegamento =  $1/R$  dove  $R$  è la velocità di trasmissione.

Se  $R = 100 \text{ Mbps}$ ,  $1 \text{ bit time} = \frac{1}{100 \cdot 10^6} = 10^{-8} \text{ s}$

# System considerations for multisegment 100BASE-T networks

Che vuol dire che 100 m di cavo UTP di categoria 5 hanno un round trip delay di 111.2 bit time (rispetto a 100BASE-T)?

111.2 bit time corrispondono a  $111.2 \times 10^{-8} \text{ s}$ . Parlando di round trip delay, si intende che in questo intervallo di tempo il segnale può attraversare il cavo avanti e indietro, percorrendo cioè 200 m.

Pertanto, la velocità di propagazione del cavo è  $v = \frac{200}{111.2 \times 10^{-8}} \text{ m/s} = 1.799 \cdot 10^8 \text{ m/s}$

Considerando la velocità della luce  $c = 299792458 \text{ m/s} \approx 3 \cdot 10^8 \text{ m/s}$

$$\frac{v}{c} = 0,6$$

Cioè la velocità di propagazione del segnale in questo mezzo è il 60% della velocità della luce.

# Livello di collegamento e LAN: tabella di marcia

- introduzione
- rilevazione e correzione degli errori
- protocolli di accesso multiplo
- **LAN**
  - indirizzamento, ARP
  - Ethernet
  - **switch**
  - VLAN
- canali virtuali: MPLS
- Reti dei data center



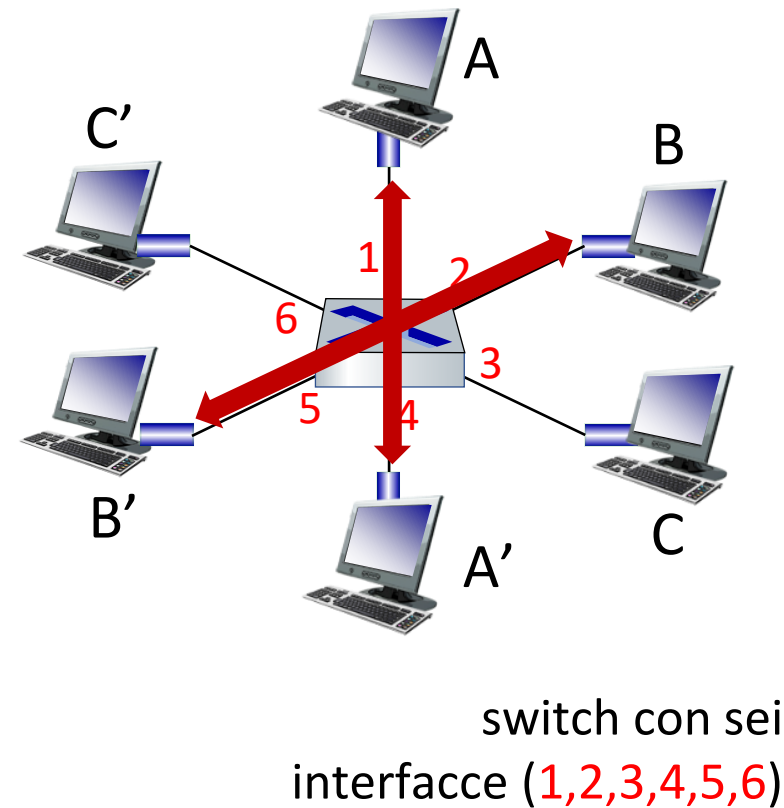
- un giorno nella vita di una richiesta web

# Switch Ethernet

- Lo switch (commutatore di pacchetti a livello di collegamento) è un dispositivo a **livello di collegamento**: ha un ruolo *attivo*
  - memorizza e inoltra (store-and-forward) frame Ethernet (o di altro tipo)
  - esamina l'indirizzo MAC di destinazione del frame in arrivo, inoltra *selettivamente* il frame in uno o più collegamenti di uscita quando il frame deve essere inoltrato in un segmento, usa CSMA/CD per accedere al segmento
- **trasparente**: gli host sono *inconsapevoli* della presenza degli switch (le cui interfacce di interconnessione agli host e router non hanno indirizzi MAC associati, o comunque non sono usati per la funzione di commutazione)
- **collegamenti eterogenei**: i collegamenti possono operare a velocità diverse e usare mezzi trasmissivi diversi; utile per evolvere la rete in maniera incrementale
- **plug-and-play, autoapprendimento**
  - non è necessario configurare gli switch

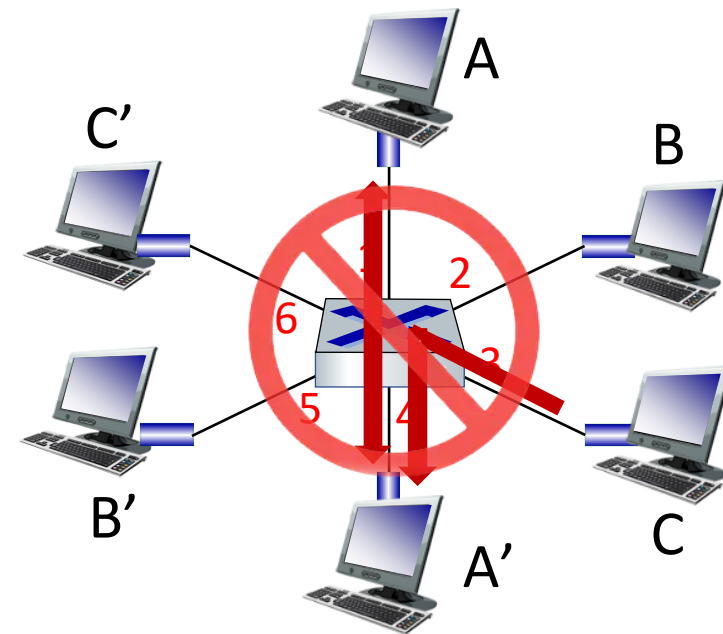
# Switch: molteplici trasmissioni simultanee

- gli host hanno connessioni dedicate, dirette con lo switch
- lo switch "bufferizza" i pacchetti
- il protocollo Ethernet è utilizzato su *ciascun* collegamento, così:
  - full-duplex: una singola coppia di nodi alle estremità del collegamento che possono trasmettere simultaneamente senza collisioni (es. perché i segnali viaggiano su fili dedicati nel cavo Ethernet), no CSMA/CD
  - half-duplex: il singolo collegamento half duplex è un dominio di collisione a sé
- **switching**: A-to-A' e B-to-B' possono trasmettere simultaneamente senza collisioni



# Switch: molteplici trasmissioni simultanee

- gli host hanno connessioni dedicate, dirette con lo switch
- lo switch "bufferizza" i pacchetti
- il protocollo Ethernet è utilizzato su *ciascun* collegamento, così:
  - full-duplex: una singola coppia di nodi alle estremità del collegamento che possono trasmettere simultaneamente senza collisioni (es. perché i segnali viaggiano su fili dedicati nel cavo Ethernet), no CSMA/CD
  - half-duplex: il singolo collegamento half duplex è un dominio di collisione a sé
- **switching**: A-to-A' e B-to-B' possono trasmettere simultaneamente senza collisioni
  - ma A-to-A' e C-to-A' *non* possono accadere simultaneamente



switch con sei  
interfacce (1,2,3,4,5,6)

# Tabella commutazione degli switch

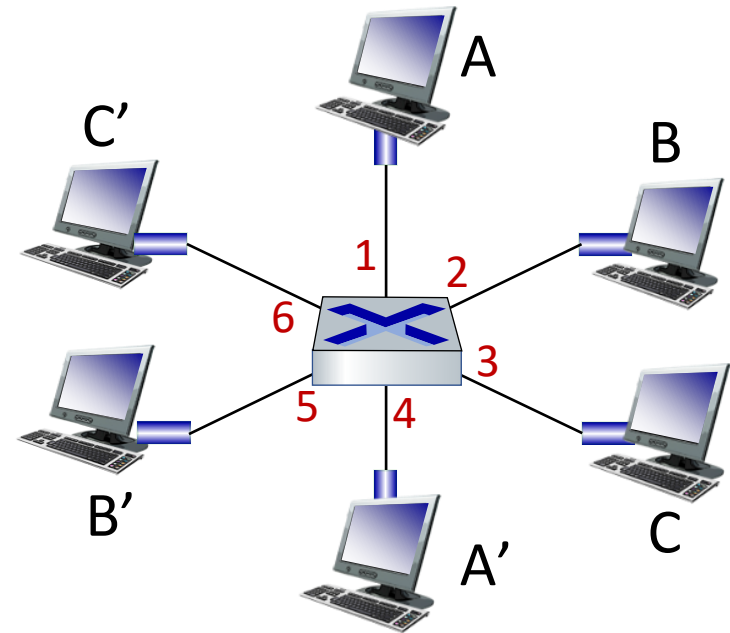
D: come sa lo switch che A' è raggiungibile tramite l'interfaccia 4, e che B' è raggiungibile dall'interfaccia 5?

R: ciascuno switch ha una **tabella di commutazione (switch table)**, ciascuna voce:

- (indirizzo MAC del nodo, interfaccia che conduce al nodo, timestamp)
- Assomiglia alle tabelle di inoltramento dei router!

D: Come vengono create e mantenute le voci nella tabella di commutazione?

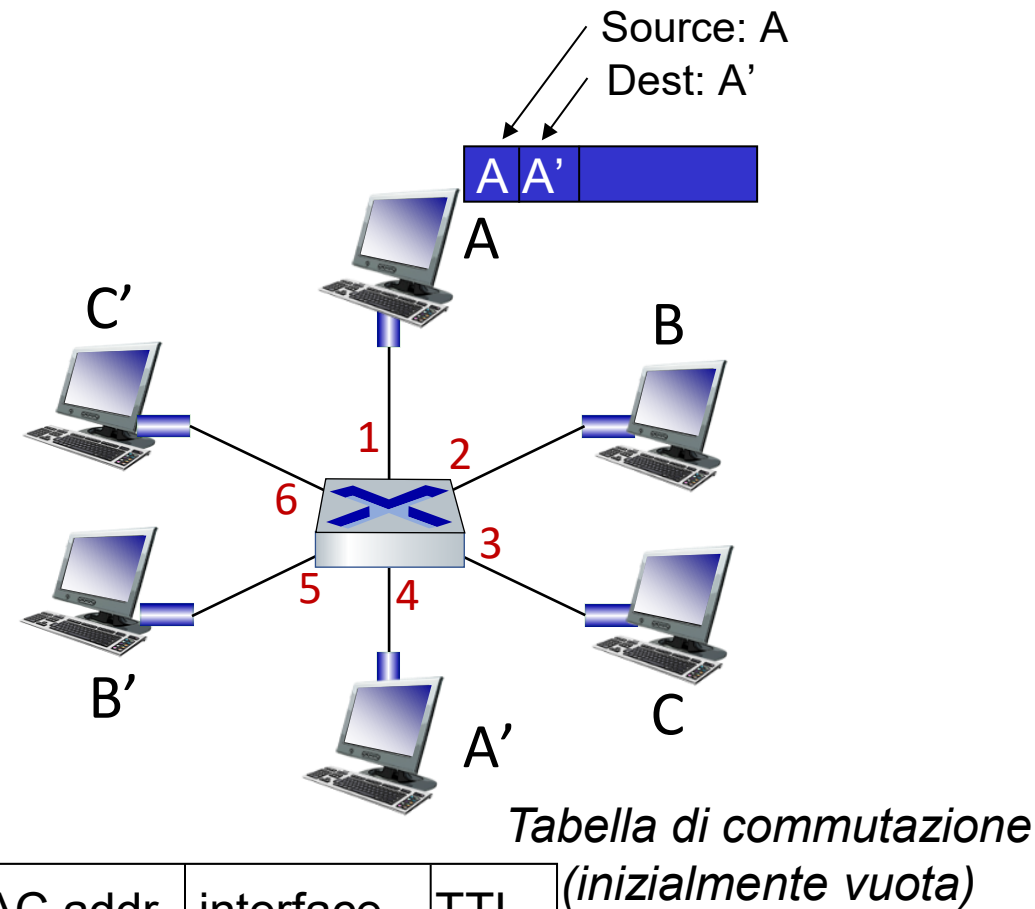
- qualcosa tipo un protocollo di instradamento?





# Switch: autoapprendimento

- uno switch *impara* quali nodo possono essere raggiunti attraverso quale interfaccia
  - quando un frame viene ricevuto, lo switch “impara” la posizione del mittente: segmento LAN in ingresso
  - registra la coppia mittente/posizione nella tabella di commutazione



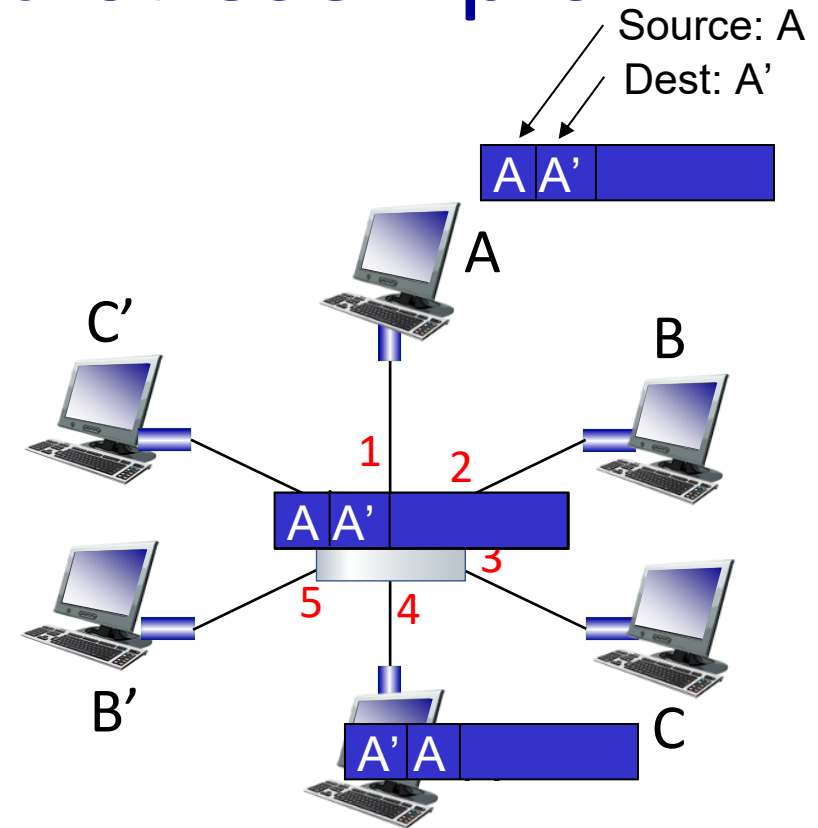
# Switch: filtraggio e inoltra dei frame

Quando uno switch riceve un frame:

1. registra il collegamento in ingresso e l'indirizzo MAC dell'host mittente
2. indicizza la tabella degli switch utilizzando l'indirizzo MAC di destinazione
3. **se** viene trovata una voce per la destinazione  
    **allora** {  
        **se** la destinazione è sul segmento dal quale è arrivato il frame  
        **allora** scarta il frame  
        **altrimenti** inoltra il frame sull'interfaccia indicata dalla voce  
    }  
    **altrimenti** flood /\* inoltra su tutte le interfacce eccetto quella di arrivo;  
                            in altre parole, manda il frame in broadcast (ma non  
                            cambia l'indirizzo MAC di destinazione) \*/

# Autoapprendimento e inoltro: esempio

- destinazione del frame, A', posizione sconosciuta: **flood**
- posizione della destinazione A conosciuta: **invia selettivamente soltanto su un collegamento**

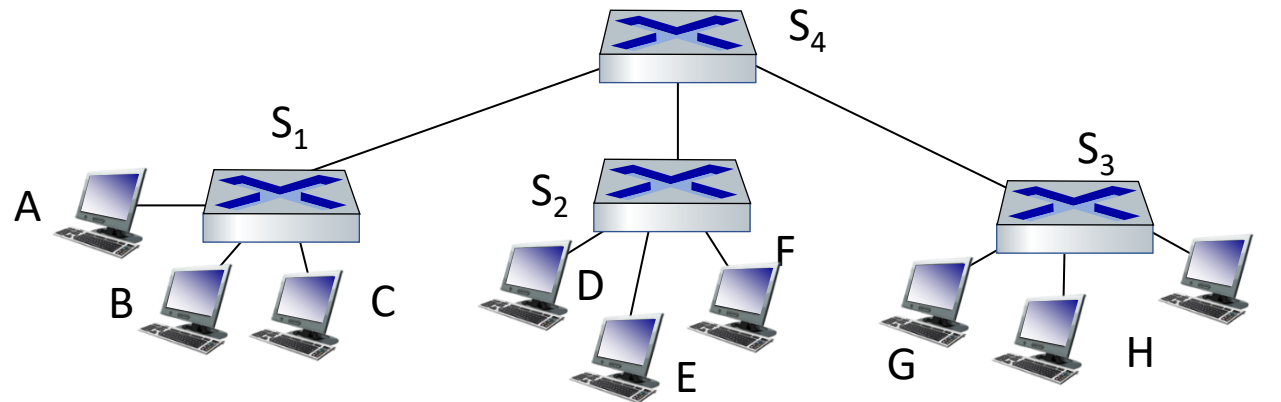


MAC addr	interface	TTL
A	1	60
A'	4	60

*tabella di commutazione /  
switch table  
(inizialmente vuota)*

# Interconnettere gli switch

gli switch con autoapprendimento possono essere interconnessi tra di loro

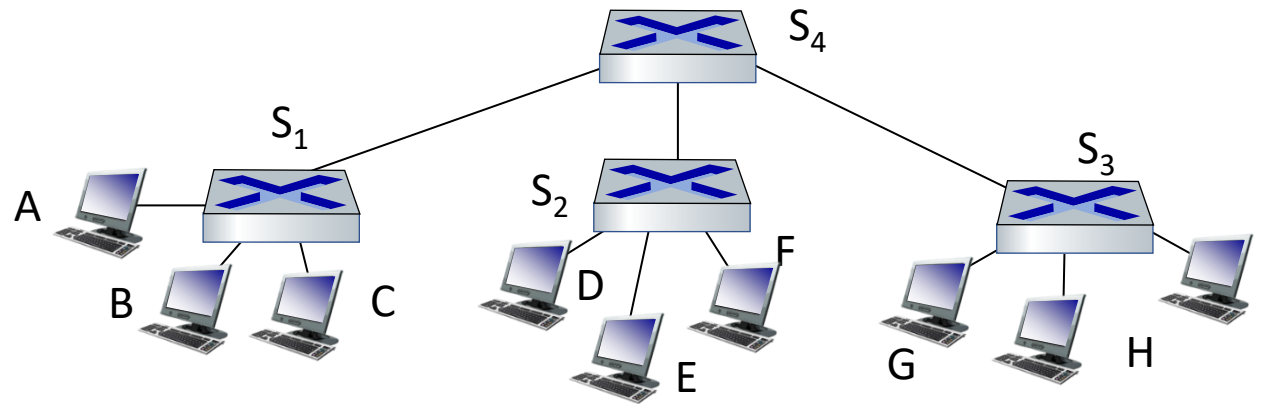


D: invio da A a G – come sa S<sub>1</sub> di inoltrare il frame destinato a G attraverso S<sub>4</sub> e S<sub>3</sub>?

- R: autoapprendimento! (funziona esattamente alla stessa maniera del caso a singolo switch!)

# Self-learning multi-switch example

Si supponga che C invii un frame a I e che I risponda a C



D: mostrare le tabelle di commutazione e l'inoltro dei pacchetti in  $S_1$ ,  $S_2$ ,  $S_3$ ,  $S_4$

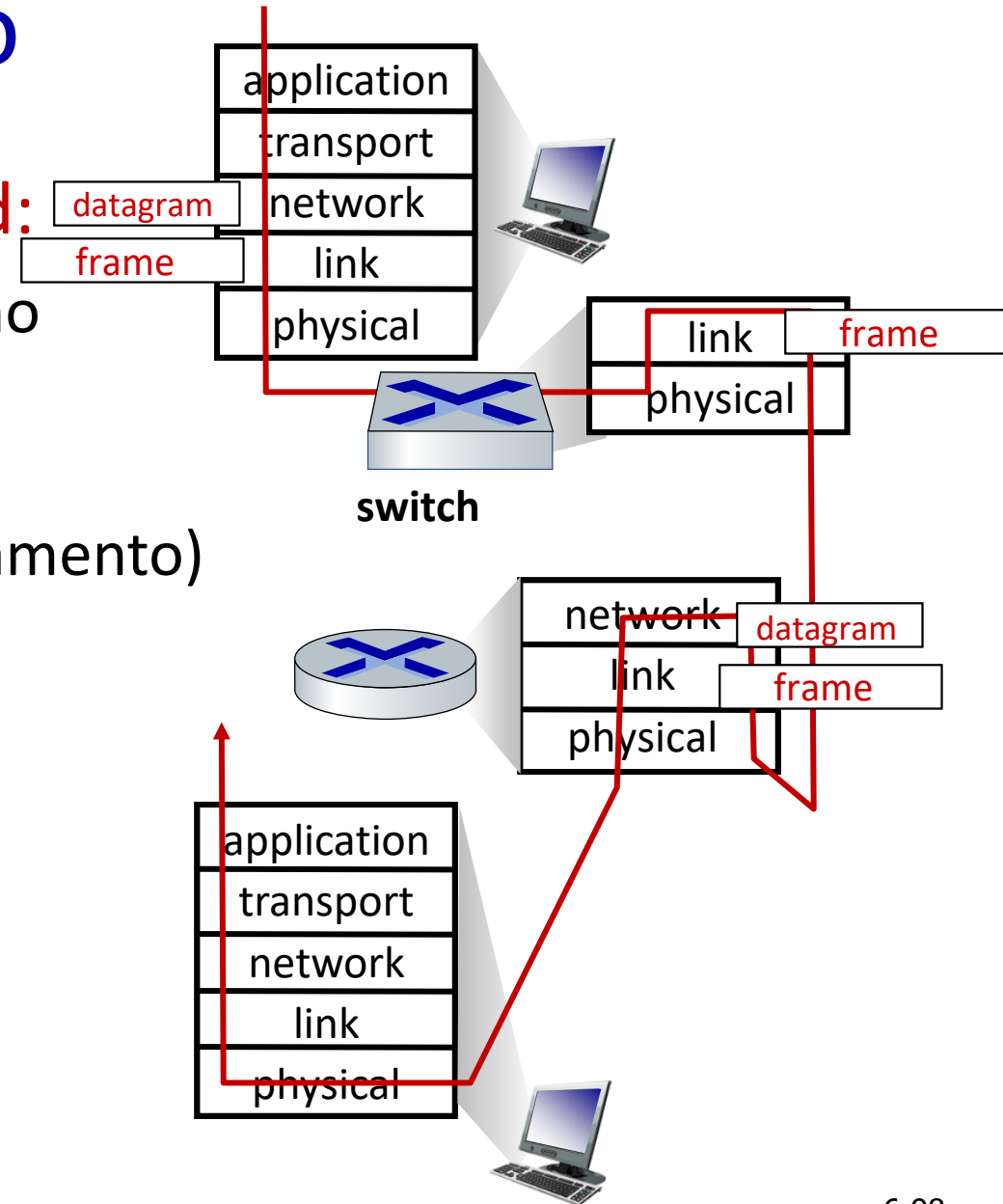
# Switch e router a confronto

Entrambi lavorano in store-and-forward:

- **router**: dispositivi a livello di rete (esaminano l'intestazione a livello di rete)
- **switch**: dispositivi a livello di collegamento (esaminano l'intestazione a livello di collegamento)

Entrambi hanno tabelle di inoltra:

- **router**: calcolano le tabelle usando algoritmi di instradamento, indirizzi IP
- **switch**: autoapprendimento della tabella di inoltra usando il flooding, indirizzi MAC



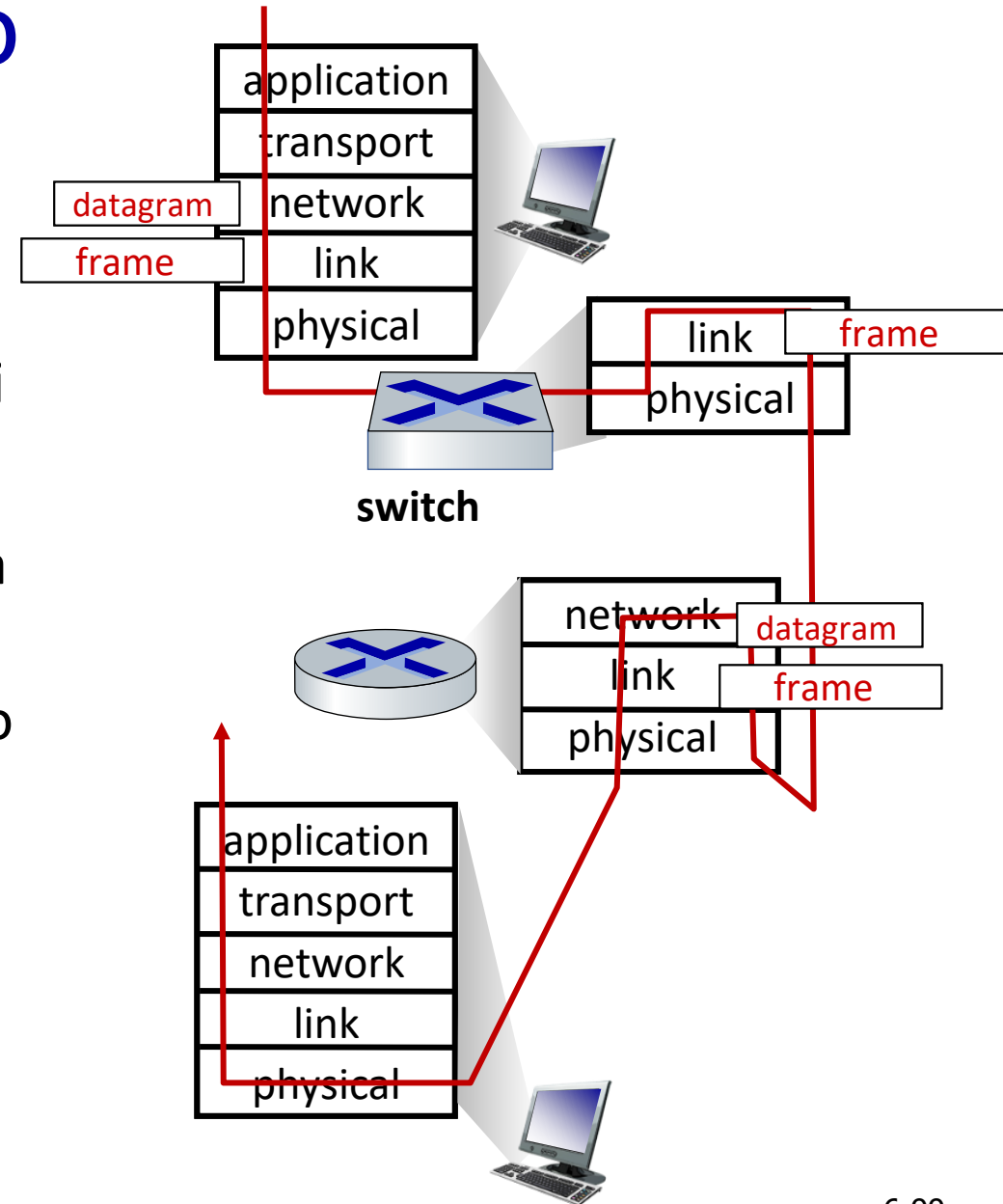
# Switch e router a confronto

## Topologia della rete:

- **router**: gli algoritmi di instradamento possono trovare percorsi ottimali (senza cicli) nonostante cicli nella topologia delle rete; inoltre, il decremento del TTL farebbe scartare i pacchetti incastrati in potenziali instradamenti ciclici (es. dovuti a errori di configurazione)
- **switch**: gli switch devono essere interconnessi a albero (anche solo logicamente, grazie al protocollo *Spanning Tree Protocol*), per evitare che il traffico broadcast (in assenza di un campo TTL nei frame) resti in circolazione potenzialmente per sempre

## Numero di nodi

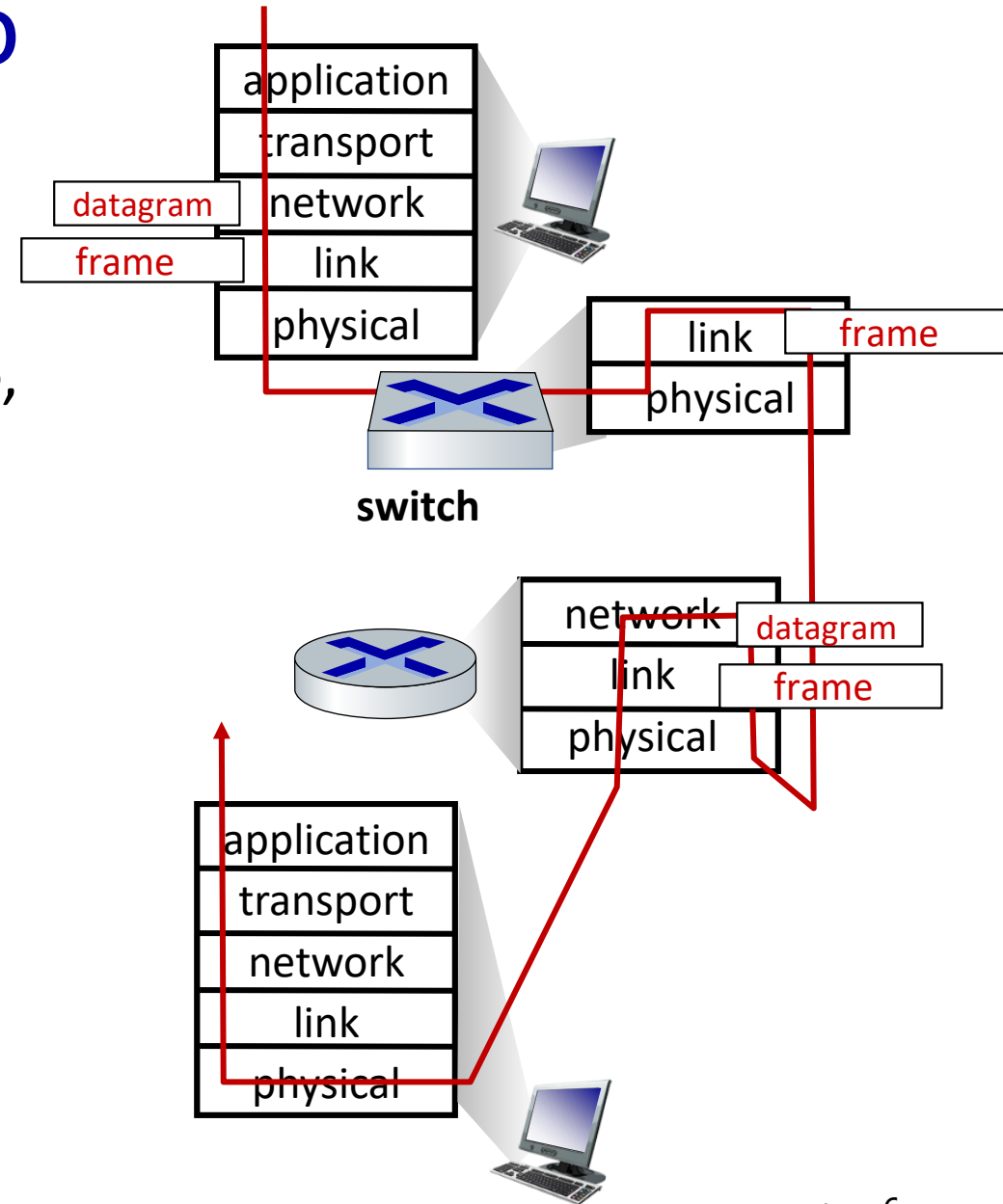
- **router**: instradamento gerarchico, aggregazione degli indirizzi, etc...
- **switch**: tabelle ARP molto grandi nei nodi, ingente traffico ARP, frame broadcast, etc...



# Switch e router a confronto

## Isolamento del traffico

- gli *switch* inviano in broadcast i frame il cui indirizzo MAC di destinazione è sconosciuto, con un effetto a valanga in presenza di molteplici switch interconnessi. I frame broadcast sono inoltrati a tutti i nodi nella rete.
- i *router* inoltrano i pacchetti in accordo a percorsi determinati dalla funzione di instradamento.





# Livello di collegamento e LAN: tabella di marcia

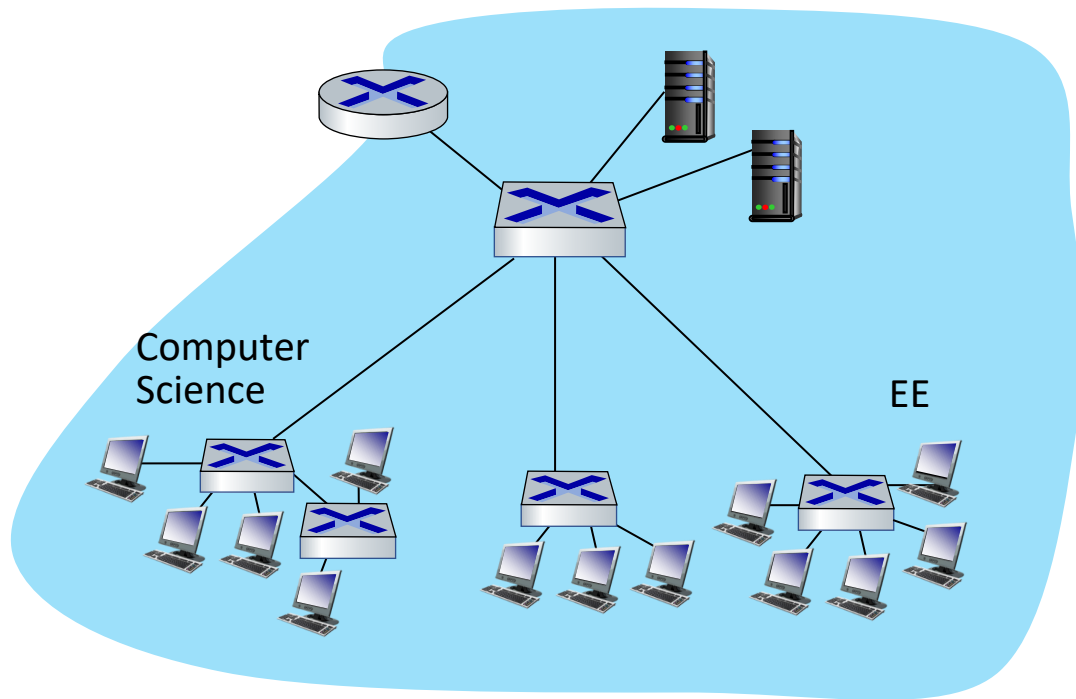
- introduzione
- rilevazione e correzione degli errori
- protocolli di accesso multiplo
- **LAN**
  - indirizzamento, ARP
  - Ethernet
  - switch
  - **VLAN**
- canali virtuali: MPLS
- Reti dei data center



- un giorno nella vita di una richiesta web

# Virtual LAN (VLAN): motivazione

*D:* Cosa succede quando le dimensioni della LAN aumentano e gli utenti cambiano il punto di attacco?

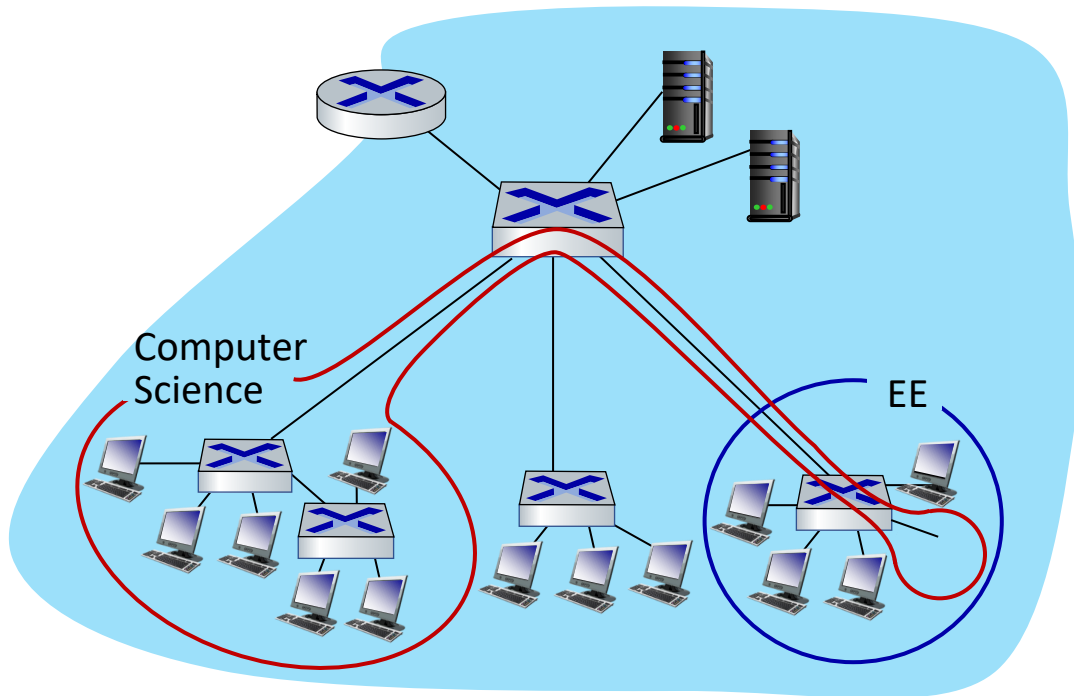


singolo dominio di broadcast:

- *scalabilità:* tutto il traffico broadcast di livello 2 (ARP, DHCP, MAC sconosciuto) deve attraversare l'intera LAN
- problemi di efficienza, sicurezza, privacy

# Virtual LAN (VLAN): motivazione

*D:* Cosa succede quando le dimensioni della LAN aumentano e gli utenti cambiano il punto di attacco?



singolo dominio di broadcast:

- *scalabilità*: tutto il traffico broadcast di livello 2 (ARP, DHCP, MAC sconosciuto) deve attraversare l'intera LAN
- problemi di efficienza, sicurezza, privacy

problemi amministrativi:

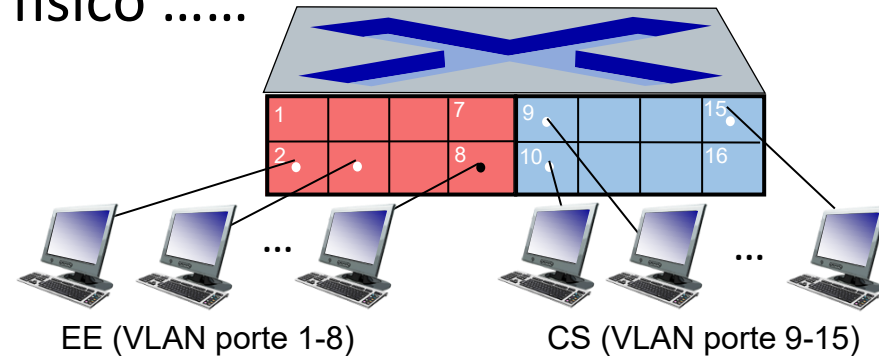
- un utente CS si sposta nell'ufficio EE - connesso *fisicamente* allo switch EE, ma vuole rimanere connesso *logicamente* allo switch CS

# VLAN basate sulle porte

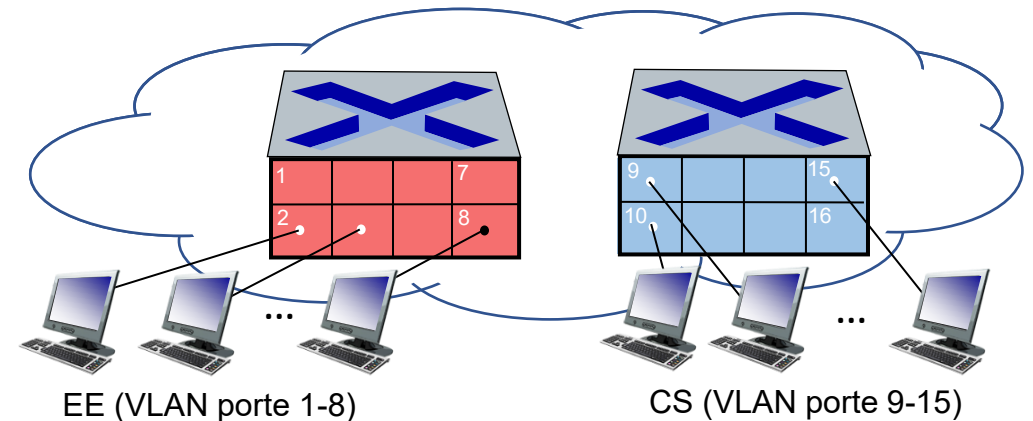
## Virtual Local Area Network (VLAN)

Gli switch che supportano le funzionalità VLAN possono essere configurati per definire più LAN *virtuali* su un'unica infrastruttura LAN fisica.

**port-based VLAN:** le porte dello switch raggruppate (tramite il software di gestione dello switch) cosicché un *singolo* switch fisico .....

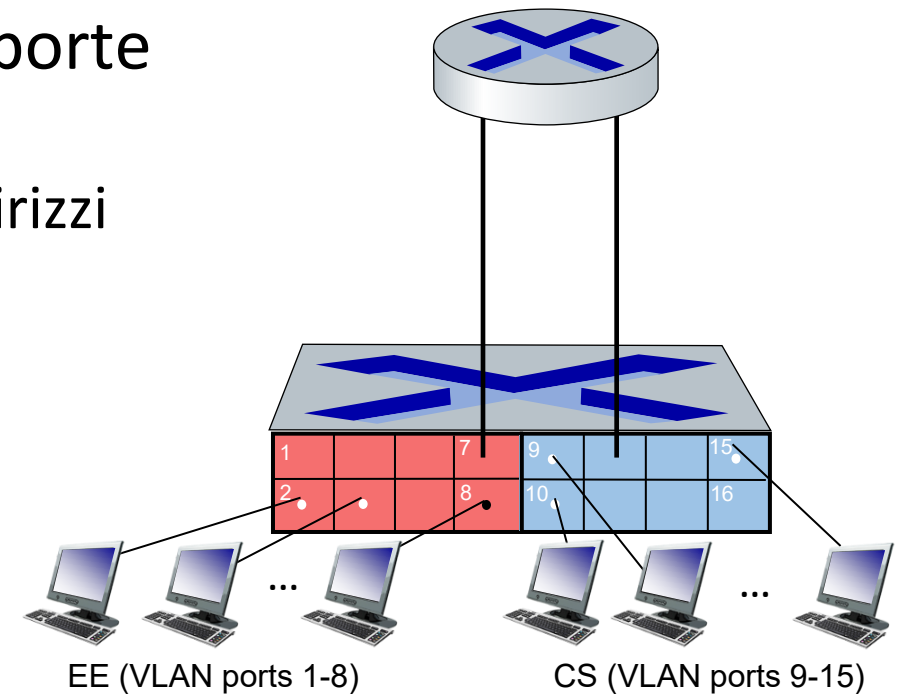


... operi come **molteplici** switch virtuali

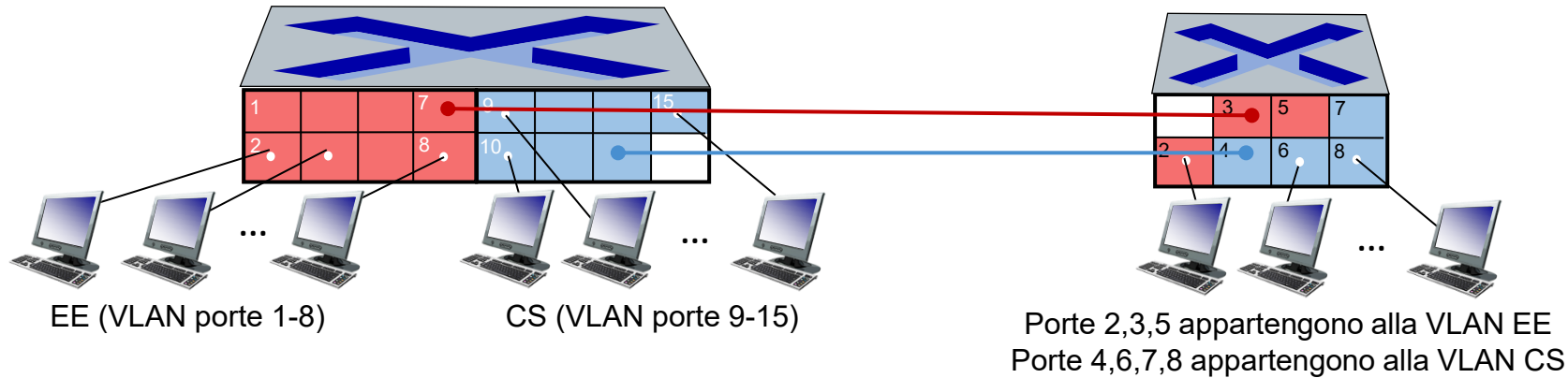


# VLAN basate sulle porte

- **isolamento del traffico:** i frame verso/da le porte 1-8 possono raggiungere *soltanto* le porte 1-8
  - Si possono definire anche VLAN basate sugli indirizzi MAC degli endpoint, piuttosto che sulle porte
- **appartenenza dinamica:** le porte possono essere assegnate dinamicamente tra le VLAN
- **inoltro tra VLAN:** fatto tramite un routing (esattamente come con switch separati)
  - in pratica i produttori combinano gli switch con i router



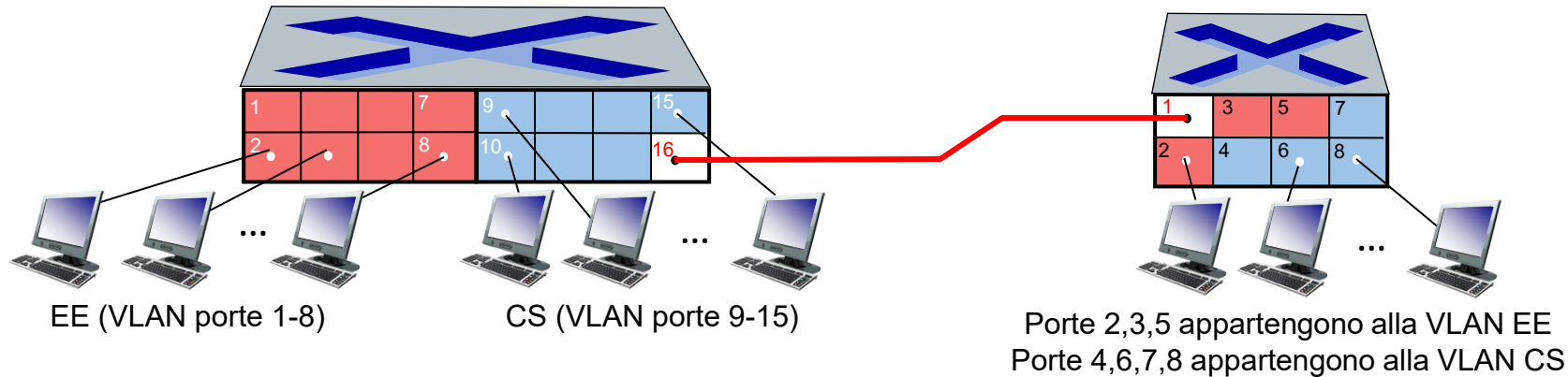
# VLAN che si estendono su più switch



## Connettere tra di loro due porte appartenenti alla stessa VLAN:

- questa soluzione *non è scalabile*: per connettere N VLAN definite su due switch fisici, dovremmo sacrificare N porte su ciascuno switch fisico

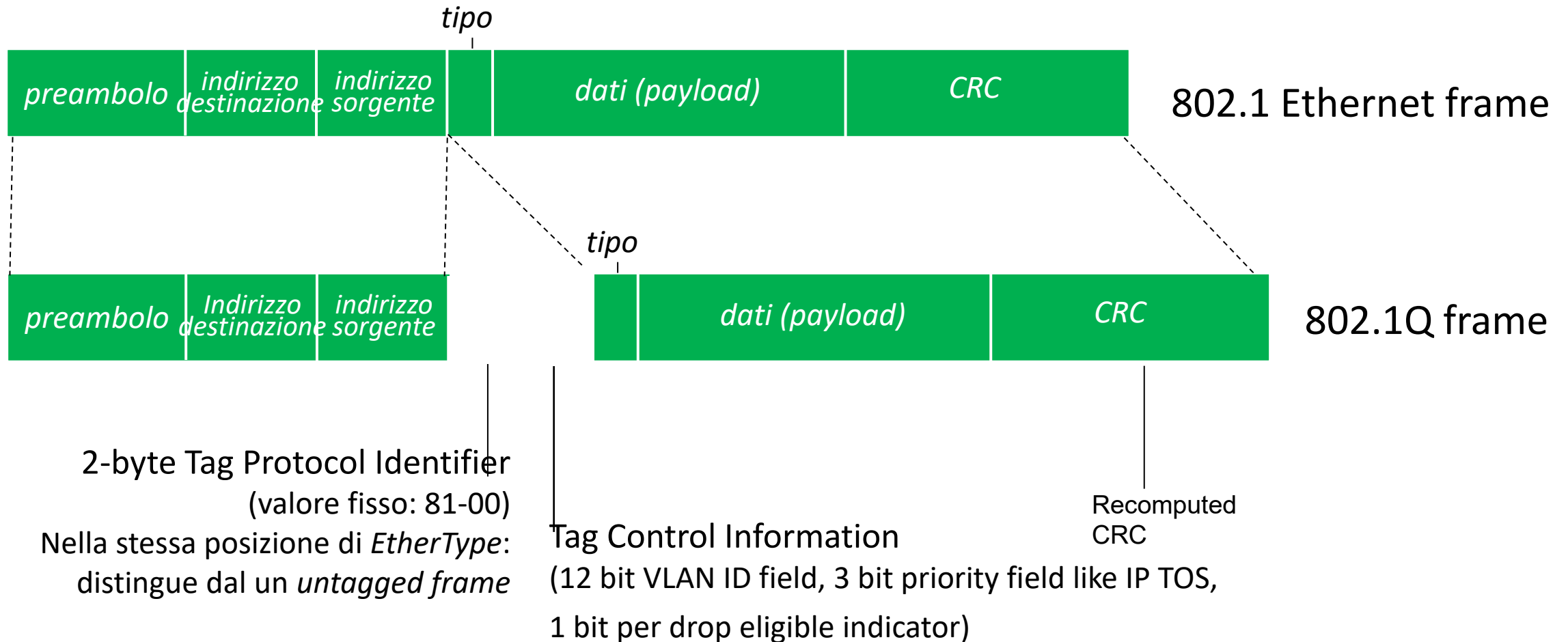
# VLAN che si estendono su più switch



**porta trunk:** trasporta frame tra VLAN definite su più switch fisici

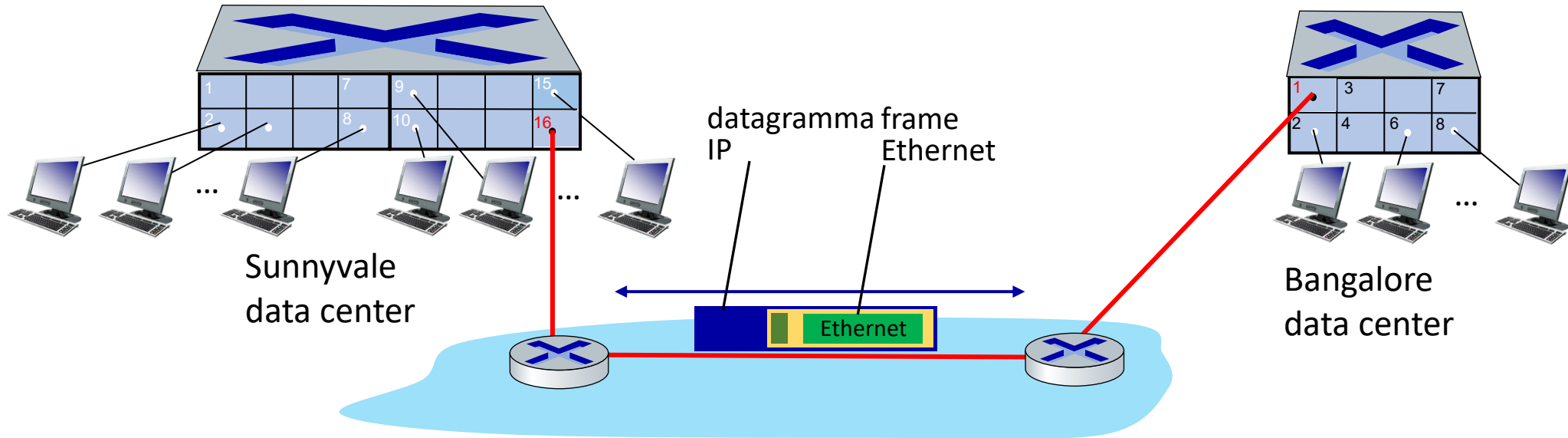
- i frame inoltrati all'interno della VLAN tra gli switch non possono essere frame vanilla 802.3 (devono contenere informazioni sull'ID VLAN)
- il protocollo 802.1q aggiunge/rimuove campi di intestazione aggiuntivi per i frame inoltrati tra le porte trunk

# Formato del frame VLAN 802.1Q





# EVPN: Ethernet VPN (altrimenti note come VXLAN)



Switch Ethernet di livello 2 connessi *logicamente* l'un l'altro (es., usando IP come *underlay*)

- frame Ethernet trasportati *dentro* a datagrammi IP tra siti
- “schema di *tunneling* per *sovrapporre reti Layer 2 a reti Layer 3* ... funziona sull'infrastruttura di rete esistente e fornisce un mezzo per “allungare” una rete Layer 2”. [RFC 7348]