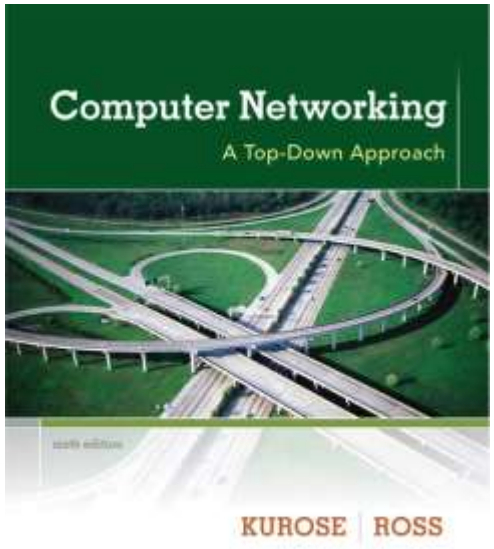
Capitolo 5
Livello di link



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

Nota per l'utilizzo:

Abbiamo preparato queste slide con l'intenzione di renderle disponibili a tutti (professori, studenti, lettori). Sono in formato PowerPoint in modo che voi possiate aggiungere e cancellare slide (compresa questa) o modificarne il contenuto in base alle vostre esigenze.

Come potete facilmente immaginare, da parte nostra abbiamo fatto un sacco di lavoro. In cambio, vi chiediamo solo di rispettare le seguenti condizioni:

- se utilizzate queste slide (ad esempio, in aula) in una forma sostanzialmente inalterata, fate riferimento alla fonte (dopo tutto, ci piacerebbe che la gente usasse il nostro libro!)
- se rendete disponibili queste slide in una forma sostanzialmente inalterata su un sito web, indicate che si tratta di un adattamento (o che sono identiche) delle nostre slide, e inserite la nota relativa al copyright.

Thanks and enjoy! JFK/KWR

Capitolo 5: Livello di link

obiettivi:

- comprendere i principi per implementare i servizi di trasmissione dati:
  - rilevazione e correzione dell'errore
  - condivisione di un canale broadcast: accesso multiplo
  - indirizzamento a livello di link
  - local area network: Ethernet, VLAN
- istanziazione e implementazione delle varie tecnologie a livello di link

Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

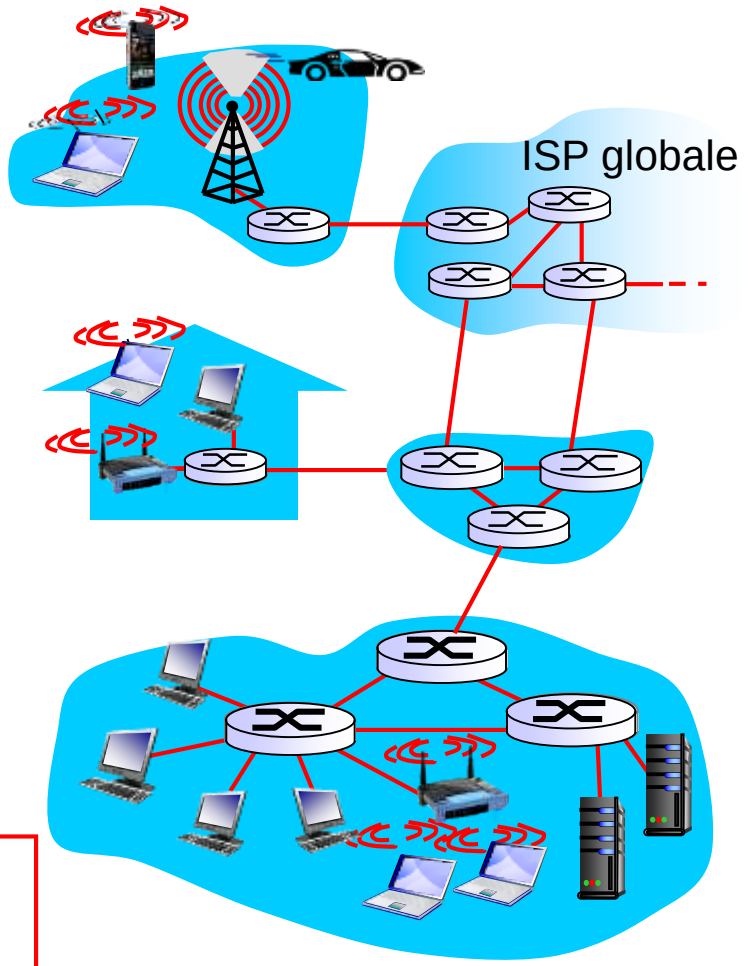
5.6 data center networking

5.7 il cammino di una richiesta web

Livello di link: introduzione

terminologia:

- host e router: nodi
- i canali di comunicazione che collegano nodi adiacenti lungo un cammino sono i collegamenti (link)
  - link cablati (wired)
  - link wireless
  - LAN
- pacchetto del layer-2: frame, che incapsula i datagrammi



il *data-link layer* ha la responsabilità di trasferire datagrammi da un nodo a un altro nodo *fisicamente adiacente* tramite un collegamento

Livello di link: contesto

- un datagramma può essere gestito da protocolli di link differenti, su link differenti:
  - es., Ethernet sul primo link, frame relay su un link intermedio, 802.11 sull'ultimo link
- ogni protocollo del livello di link fornisce servizi differenti
  - es., può fornire o non fornire il trasporto affidabile lungo il link

analogia con un viaggio:

- viaggio da Princeton a Losanna
  - taxi: da Princeton al JFK
  - aereo: dal JFK a Ginevra
  - treno: da Ginevra a Losanna
- turista = datagramma
- tratta del viaggio = link di comunicazione
- tipologia del trasporto = protocollo del livello di link
- agente di viaggio = algoritmo di routing

Servizi del livello di link

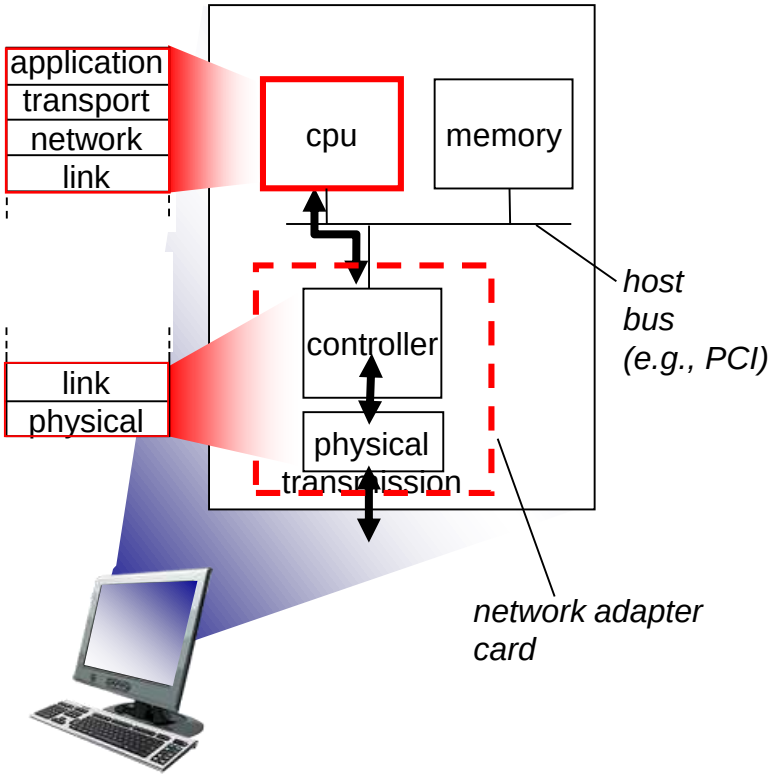
- *framing:*
  - i protocolli incapsulano i datagrammi del livello di rete all'interno di un frame a livello di link
  - regolano l'accesso al canale se il mezzo è condiviso
  - gli indirizzi "MAC" sono usati negli header dei frame per identificare sorgente e destinazione
    - differenti dagli indirizzi IP!
- *consegna affidabile tra nodi adiacenti*
  - abbiamo visto i principi (capitolo 3)!
  - usata raramente sui link a basso tasso di errore (fibra, alcuni tipi di cavi in rame)
  - link wireless: alto tasso di errore
    - *D:* perché avere affidabilità sia a livello di link che nell'end-to-end?

Servizi del livello di link (continua)

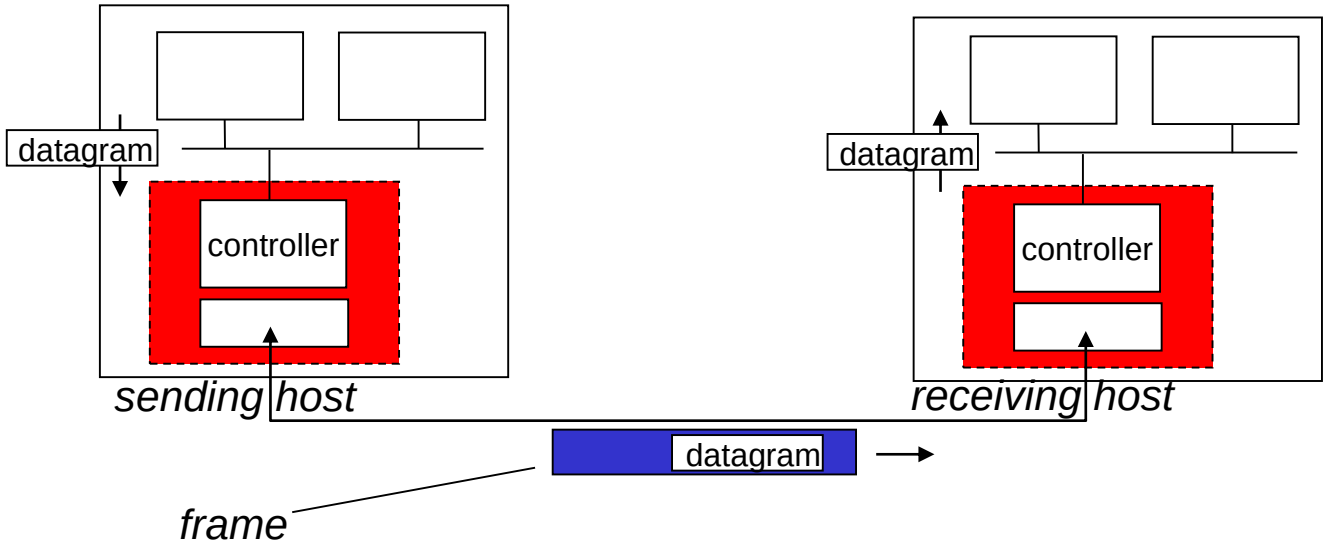
- *controllo di flusso:*
  - regola la velocità di invio tra nodi adiacenti
- *rilevazione degli errori:*
  - errori causati dall'attenuazione del segnale o da rumore elettromagnetico.
  - il nodo ricevente individua la presenza di errori:
    - può richiedere la ritrasmissione al nodo sorgente o scartare il frame
- *correzione degli errori:*
  - il ricevente identifica *e corregge* errori nei bit senza necessità di ritrasmissioni
- *half-duplex e full-duplex*
  - con half duplex, i nodi uniti da un link non possono trasmettere contemporaneamente uno verso l'altro; con full-duplex sì

Dov'è implementato il link layer?

- in tutti gli host
- il link layer è implementato negli "adattatori" (cioè le *network interface card* NIC) o su un chip
  - Ethernet card, 802.11 card; Ethernet chipset
  - implementano sia il layer link che quello fisico
- si collegano ai bus di sistema degli host
- combinazione di hardware, software, firmware



Comunicazione tra adattatori



- lato invio:
  - incapsula i datagrammi in frame
  - aggiunge i bit di error checking, rdt, controllo di flusso, etc.
- lato ricezione
  - controlla gli errori, rdt, controllo di flusso, etc
  - estrae i datagrammi, e li passa ai livelli superiori

Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

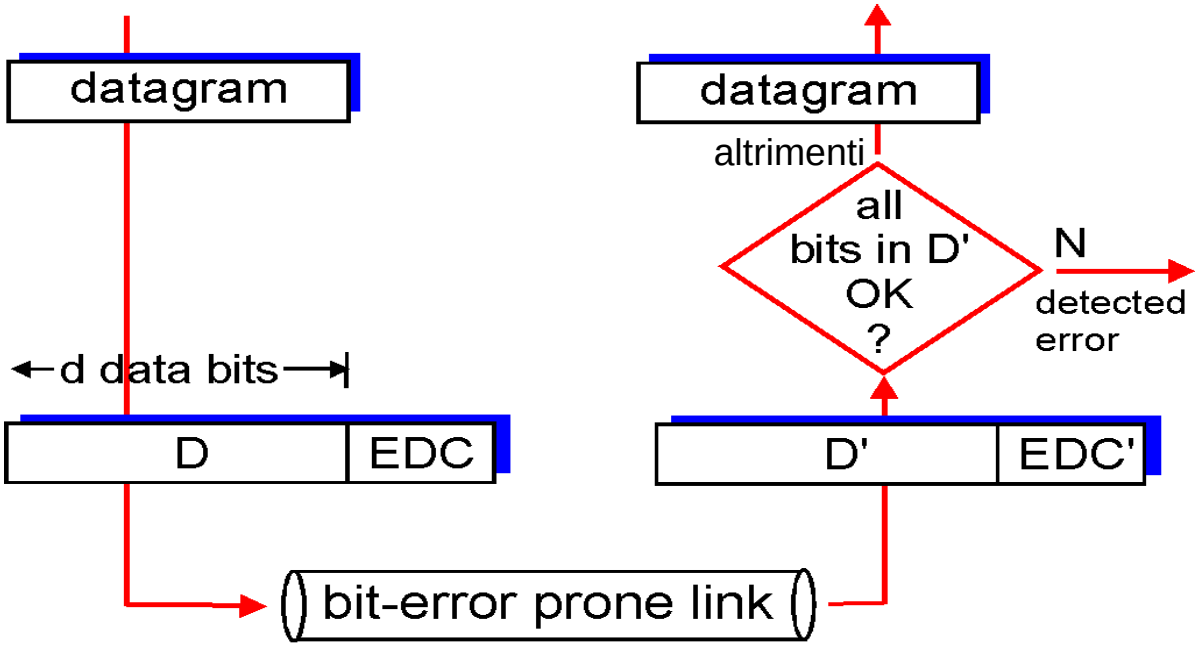
5.6 data center networking

5.7 il cammino di una richiesta web

Rilevazione degli errori

EDC= Error Detection and Correction (ridondanza)
D = Dati protetti dall'error checking, possono includere delle intestazioni

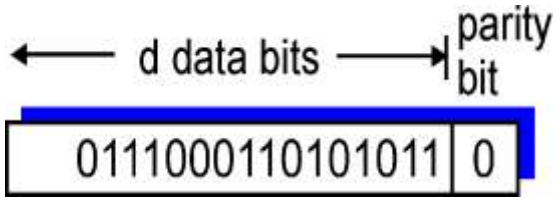
- La rilevazione degli errori non è attendibile al 100%!
  - è possibile che ci siano errori non rilevati
  - campi EDC più grandi riducono questa possibilità



Controllo di parità

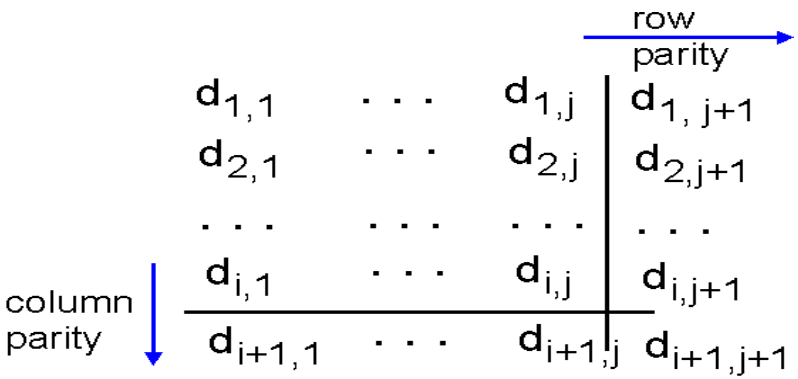
bit di parità singolo:

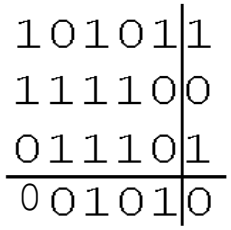
- rileva l'errore in un bit



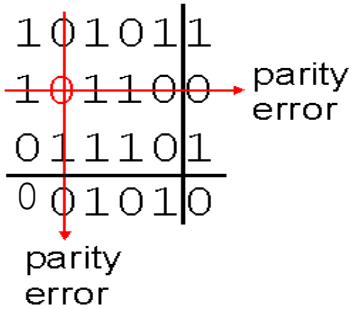
bit di parità bi-dimensionale:

- rileva e *corregge* il bit alterato





no errors



correctable single bit error

Checksum in Internet (richiamo)

obiettivo: rileva gli "errori" (es., bit alterati) nei pacchetti trasmessi (nota: usato *solo* a livello di trasporto)

mittente:

- i dati sono trattati come interi a 16 bit e sommati
- checksum: è il complemento a 1 di questa somma
- il mittente inserisce il valore della checksum nell'intestazione dei segmenti

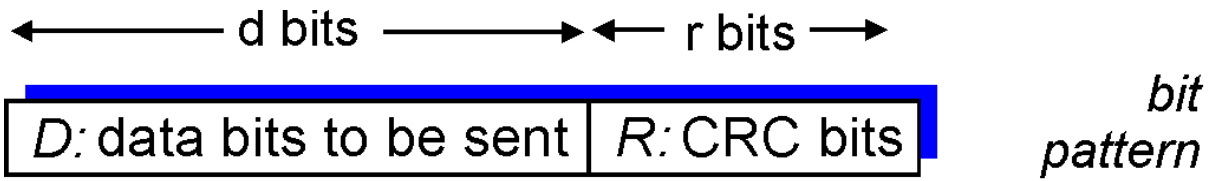
destinatario:

- calcola la checksum del segmento ricevuto
- controlla se è uguale al campo checksum :
  - NO – errore rilevato
  - SÍ - nessun errore rilevato.

  potrebbero esserci comunque errori?

Controllo a ridondanza ciclica

- error-detection più potente
- vede i bit dati, D, come un numero binario
- sceglie un stringa di r+1 bit (generatore), G
- obiettivo: sceglie r bit CRC, R, tali che
  - <D,R> siano esattamente divisibili per G (modulo 2)
  - il destinatario conosce G, divide <D,R> per G. Se il resto è diverso da 0 si è verificato un errore!
  - può rilevare errori a raffica inferiori a r+1 bit
- nella pratica è molto usato (Ethernet, 802.11 WiFi, ATM)



$D * 2^r \quad XOR \quad R$ — *mathematical formula*

Esempio di CRC

vogliamo:

$D \cdot 2^r$ XOR R = nG

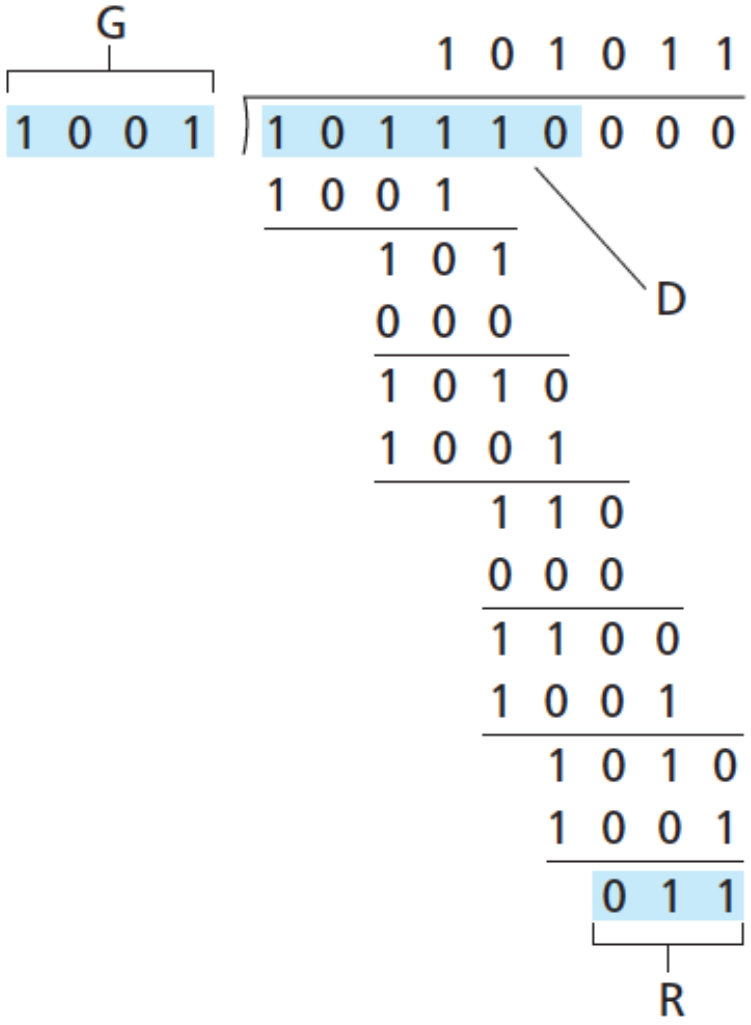
ovvero:

$D \cdot 2^r$ = nG XOR R

quindi:

se dividiamo $D \cdot 2^r$ per G, vogliamo che R soddisfi:

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



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

5.6 data center networking

5.7 il cammino di una richiesta web

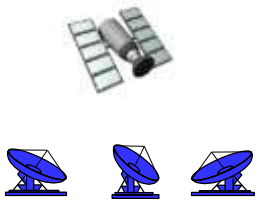
Protocolli di accesso multiplo

due tipi di "link":

- punto punto
  - PPP (point-to-point protocol) per connessioni telefoniche
  - link punto-punto tra switch Ethernet e host
- *broadcast (cavo o canale condiviso)*
  - Ethernet tradizionale
  - HFC in upstream
  - 802.11 wireless LAN

cavo condiviso(es., Ethernet cablata)

RF condivisa (es., 802.11 WiFi)



RF condivisa (satellite)

uomini a un cocktail party (aria e acustica condivisa)

Protocolli di accesso multiplo

- singolo canale broadcast condiviso
- sono possibili due o più trasmissioni simultanee da parte dei nodi: interferenza
  - *collisione* se un nodo riceve due o più segnali contemporaneamente

protocollo di accesso multiplo

- algoritmo distribuito che determina le modalità con cui i nodi regolano le loro trasmissioni sul canale condiviso
- la comunicazione relativa alla condivisione del canale deve utilizzare lo stesso canale!
  - non c'è un canale "out-of-band" per la coordinazione

Protocolli di accesso multiplo ideale

dato: canale broadcast con rate R bps

ideale:

1. quando un nodo deve inviare dati, questo dispone di un tasso trasmissivo pari a R bps.
2. quando M nodi devono inviare dati, questi dispongono di un tasso trasmissivo pari a R/M bps
3. totalmente decentralizzato:
   - non ci sono nodi speciali per coordinare le trasmissioni
   - non c'è sincronizzazione di clock o di slot
4. semplice

Protocolli MAC: classificazione

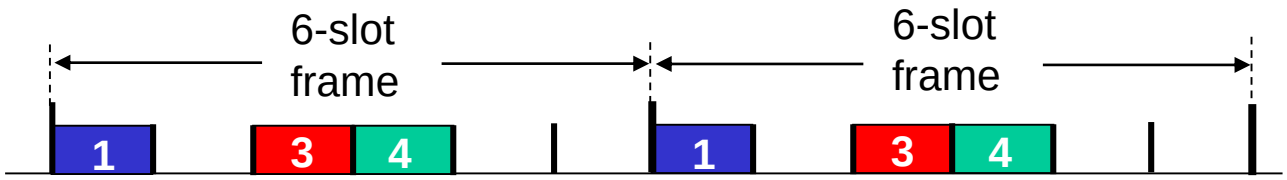
tre categorie principali:

- *suddivisione del canale*
  - si suddivide un canale in "parti più piccole" (slot di tempo, frequenza, codice).
  - alloca la parte a un nodo per un uso esclusivo
- *accesso casuale*
  - i canali non vengono divisi e si possono verificare collisioni
  - meccanismi di "recupero" a seguito di una collisione
- *a rotazione ("taking turns")*
  - ciascun nodo ha il suo turno di trasmissione, ma i nodi che hanno molto da trasmettere possono avere turni più lunghi

Protocolli a suddivisione del canale: TDMA

TDMA: time division multiple access

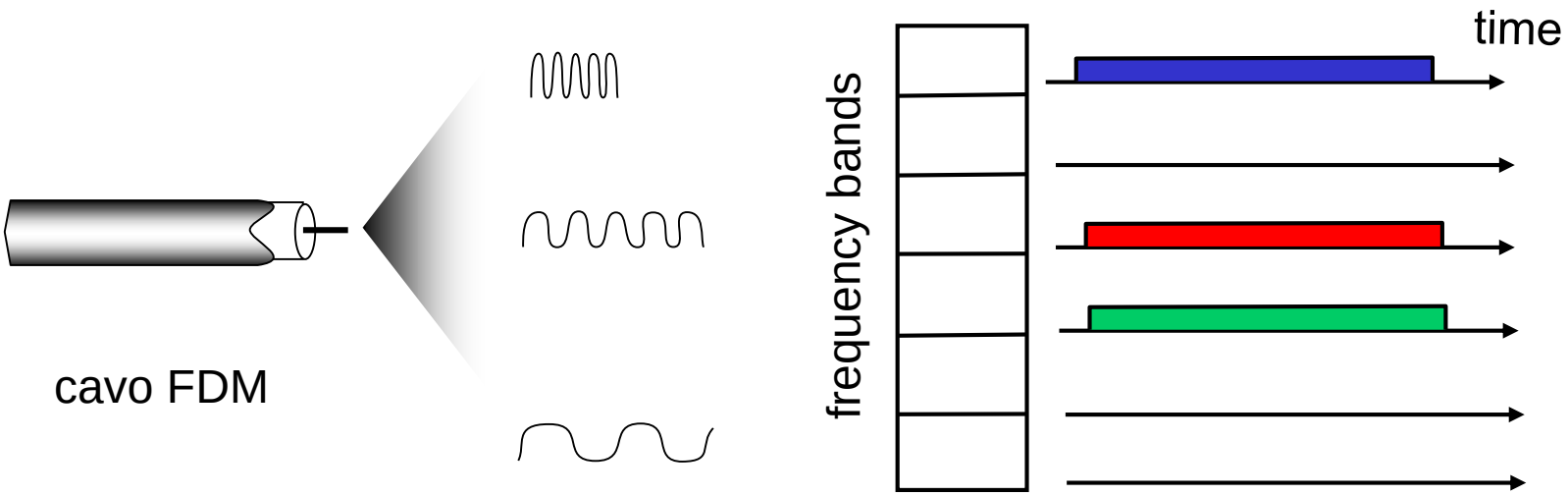
- accesso al canale a “turni"
- ogni nodo ottiene uno slot di lunghezza fissata (lunghezza = tempo per trasmettere pacchetti) a ogni turno
- gli slot non usati rimangono inattivi
- esempio: 6-nodi in una LAN, 1,3,4 hanno pacchetti, slot 2,5,6 inutilizzati



Protocolli a suddivisione del canale: FDMA

FDMA: frequency division multiple access

- canale suddiviso in bande di frequenza
- ogni nodo ha una banda di frequenza assegnata
- bande di frequenza possono rimanere inutilizzate
- esempio: 6-nodi in una LAN, 1,3,4 hanno pacchetti, bande di frequenza 2,5,6 inutilizzate



Protocolli ad accesso casuale

- quando un nodo deve inviare un pacchetto
  - trasmette al massimo rate R consentito dal canale
  - non vi è coordinazione *a priori* tra i nodi
- se due o più nodi trasmettono insieme ➜ "collisione"
- un protocollo ad accesso casuale specifica:
  - come rilevare un'eventuale collisione
  - come recuperare da una collisione (es., tramite una ritrasmissione ritardata)
- esempi di protocolli ad accesso casuale:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA

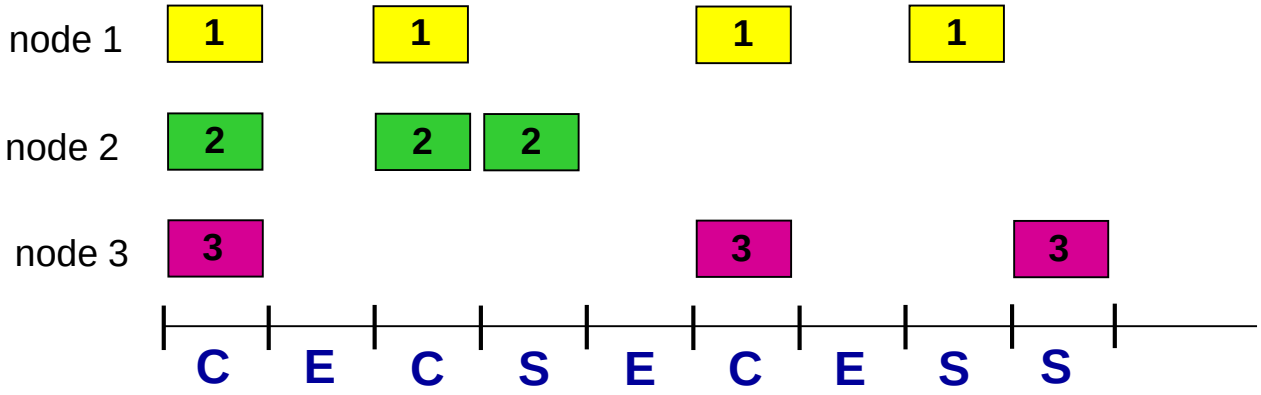
assunzioni:

- tutti i pacchetti hanno la stessa dimensione
- tempo suddiviso in slot; ogni slot equivale al tempo di trasmissione di un pacchetto
- i nodi iniziano la trasmissione dei pacchetti solo all'inizio degli slot
- i nodi sono sincronizzati
- se 2 o più nodi trasmettono nello stesso slot, tutti i nodi rilevano la collisione

operazioni:

- quando un nodo ha un nuovo pacchetto da spedire, il nodo attende fino all'inizio dello slot successivo
  - *se non c'è collisione:* il nodo può trasmettere un nuovo pacchetto nello slot successivo
  - *se c'è collisione:* il nodo ritrasmette con probabilità *p* il suo pacchetto durante gli slot successivi

Slotted ALOHA



Pro:

- un singolo nodo può trasmettere continuamente pacchetti alla massima velocità del canale
- è fortemente decentralizzato: solo gli slot devono essere sincronizzati
- semplice

Contro:

- le collisioni sprecano gli slot
- slot inutilizzati
- i nodi possono rilevare la collisione prima del fine slot
- sincronizzazione necessaria

Slotted ALOHA: efficienza

efficienza: frazione di slot con successo (con molti nodi, che hanno molti pacchetti da spedire)

- *supponiamo: N* nodi con pacchetti da spedire, ognuno trasmette i pacchetti in uno slot con probabilità *p*
- probabilità di successo di un dato nodo = $p(1-p)^{N-1}$
- probabilità che *ogni* nodo abbia successo = $Np(1-p)^{N-1}$

- massima efficienza: trovare p* che massimizza $Np(1-p)^{N-1}$
- per un elevato numero di nodi, ricaviamo il limite di $Np^*(1-p^*)^{N-1}$ per N che tende all'infinito, e otteniamo:

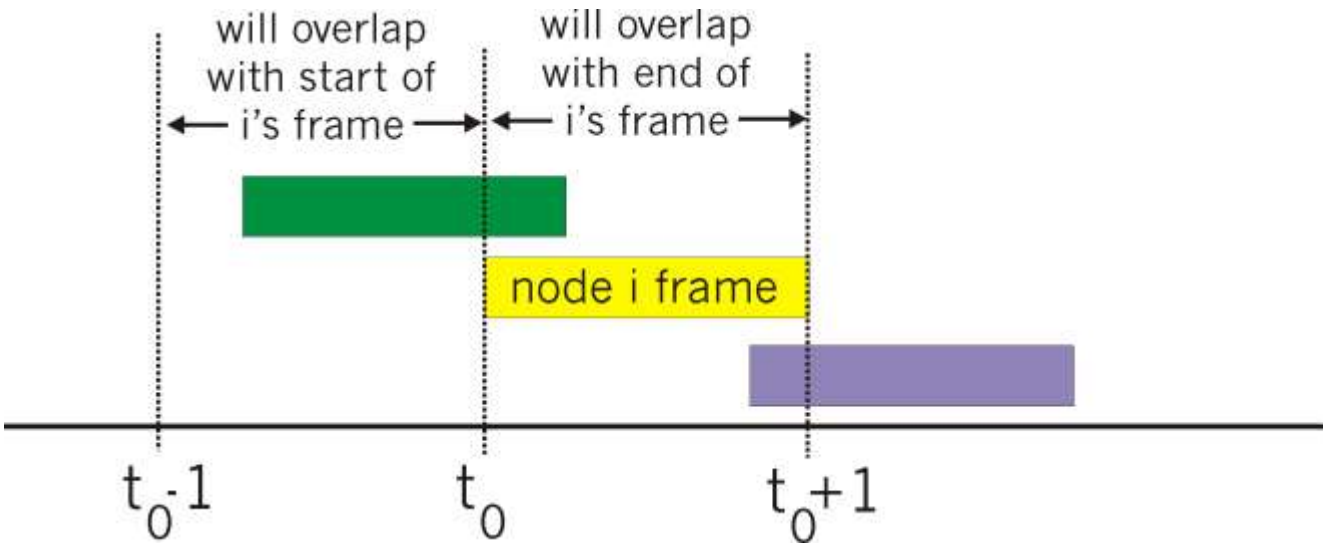
efficienza massima = 1/e = .37

nel caso migliore: solo il 37% degli slot compie lavoro utile!

!

ALOHA puro (unslotted)

- unslotted Aloha: più semplice, non sincronizzato
- quando arriva il primo pacchetto
  - lo trasmette immediatamente
- la probabilità di collisione cresce:
  - il pacchetto trasmesso al t_0 collide con altri pacchetti trasmessi nell'intervallo $[t_0-1, t_0+1]$



ALOHA puro: efficienza

P(successo per un dato nodo) = P(il nodo trasmette) ·
P(nessun altro nodo trasmette in $[t_0\text{-}1, t_0]$) ·
P(nessun altro nodo trasmette in $[t_0\text{-}1, t_0]$)

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

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

… scegliendo p migliore e con $n \rightarrow \infty$

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

peggio dello slotted Aloha!

CSMA (carrier sense multiple access)

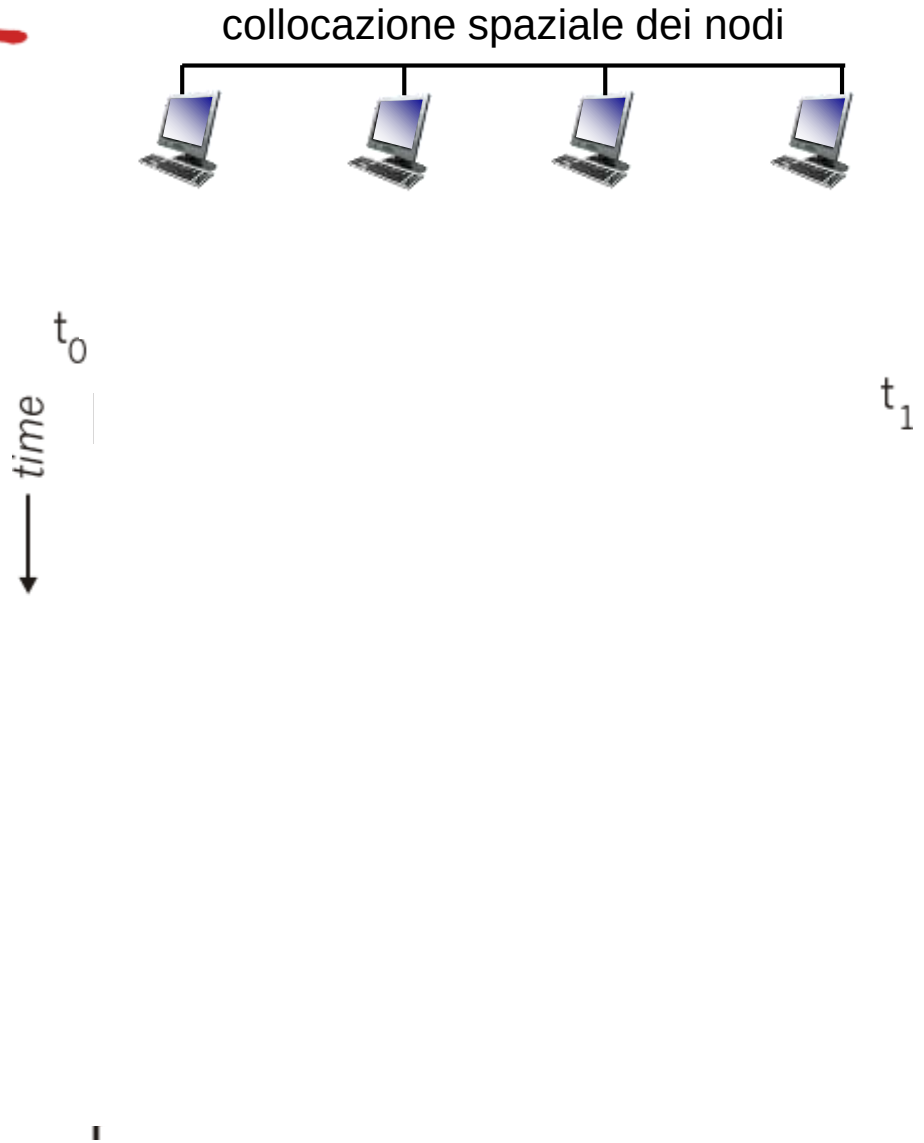
CSMA: ascolta prima di trasmettere:

- se il canale risulta libero: trasmette l'intero pacchetto
- se il canale risulta occupato: ritarda la trasmissione

- analogia umana: non interrompere chi parla!

Collisioni CSMA

- Le collisioni *possono* ancora verificarsi: il ritardo di propagazione fa sì che due nodi non rilevino la reciproca trasmissione
- collisione: tutto il tempo di trasmissione del pacchetto è sprecato
  - la distanza e il ritardo di propagazione giocano un ruolo importante nel determinare la probabilità di collisione



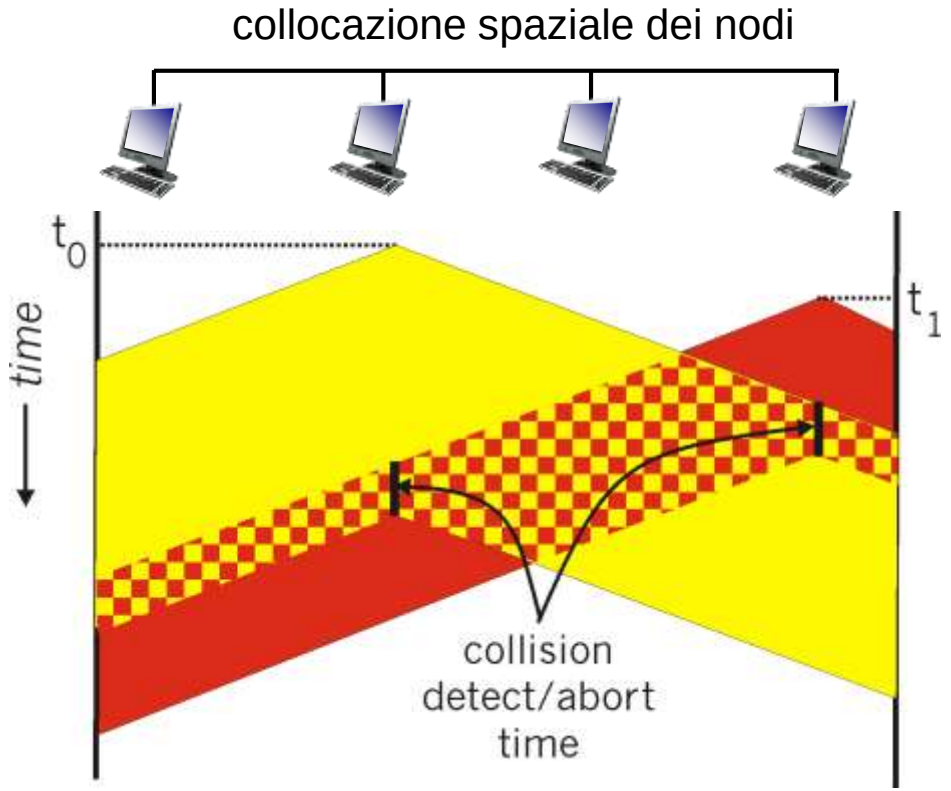
CSMA/CD (collision detection)

CSMA/CD: rilevamento della portante, differimento come in CSMA

- le collisioni vengono *rilevate* in breve tempo
- rilevata la collisione , la trasmissione viene interrotta

- rilevazione della collisione:
  - facile nelle LAN cablate: misura le potenze dei segnali
  - difficile nelle LAN wireless: la potenza dei segnali ricevuti è molto inferiore a quella della trasmissione locale
- analogia umana: un interlocutore educato

CSMA/CD (collision detection)



Algoritmo CSMA/CD in Ethernet

1. La NIC riceve il datagramma dal livello di network, crea il frame
2. Se la NIC rileva che il canale è libero, inizia la trasmissione del frame. Se rileva che il canale è occupato, aspetta fino a che si libera, e poi trasmette.
3. Se la NIC trasmette l'intero frame senza rilevare altre trasmissioni, il suo compito è finito!
4. Se rileva un'altra trasmissione mentre sta trasmettendo, interrompe e invia un segnale di jam
5. Dopo l'interruzione, la NIC parte con il *binary (exponential) backoff:*
   - dopo la *m*-esima collisione, la NIC sceglie *K* a caso in $\{0, 1, 2, \ldots, 2^m-1\}$. Aspetta $K \cdot 512$ tempi bit, e torna al passo 2
   - intervalli di backoff più lunghi con più collisioni

Efficienza CSMA/CD

- t_{prop} = massimo ritardo di propagazione tra 2 nodi nella LAN
- t_{trans} = tempo di trasmissione di un frame di dimensione massima

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

- l'efficienza tende a 1
  - per t_{prop} che tende a 0
  - per t_{trans} che tende a infinito
- prestazioni migliori di ALOHA: inoltre semplice, economico e decentralizzato!

Protocolli "a rotazione"

protocolli a partizionamento del canale:

- condividono il canale *equamente* ed *efficientemente* con carichi elevati
- inefficienti con carichi non elevati

protocolli ad accesso casuale

- efficienti con carichi non elevati: un singolo nodo può utilizzare interamente il canale
- con carichi elevati: eccesso di collisioni

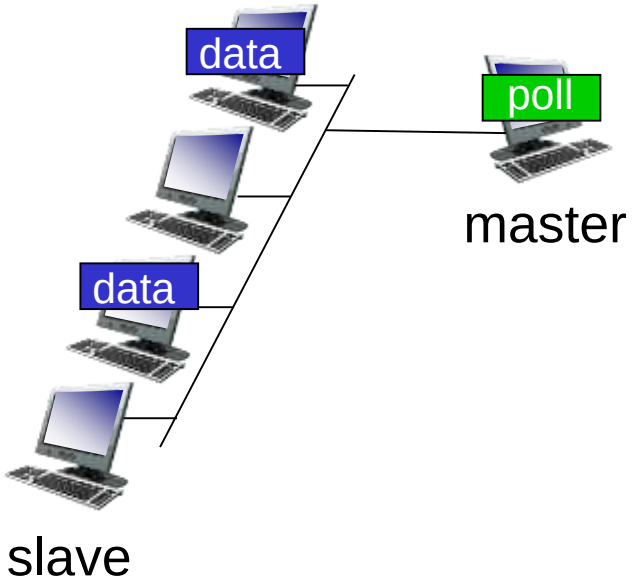
protocolli "a rotazione"

prendono il meglio dei due protocolli!

Protocolli "a rotazione"

polling:

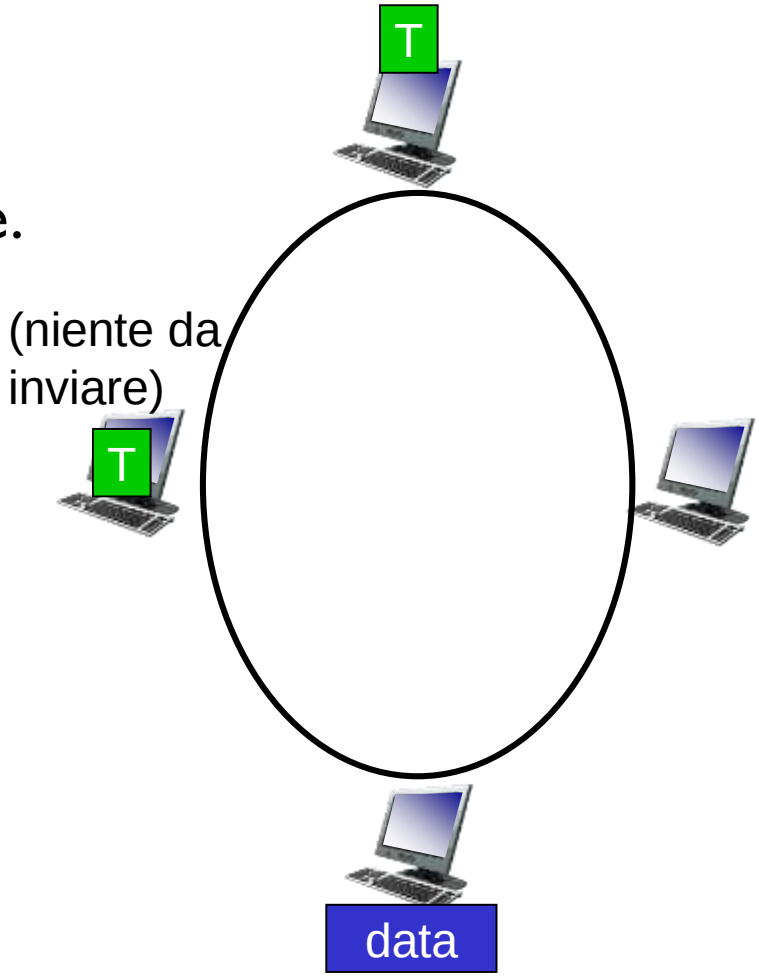
- un nodo master "invita" i nodi slave a trasmettere a turno
- usato tipicamente con dispositivi slave "stupidi"
- difetti:
  - overhead per il polling
  - latenza
  - single point of failure (master)



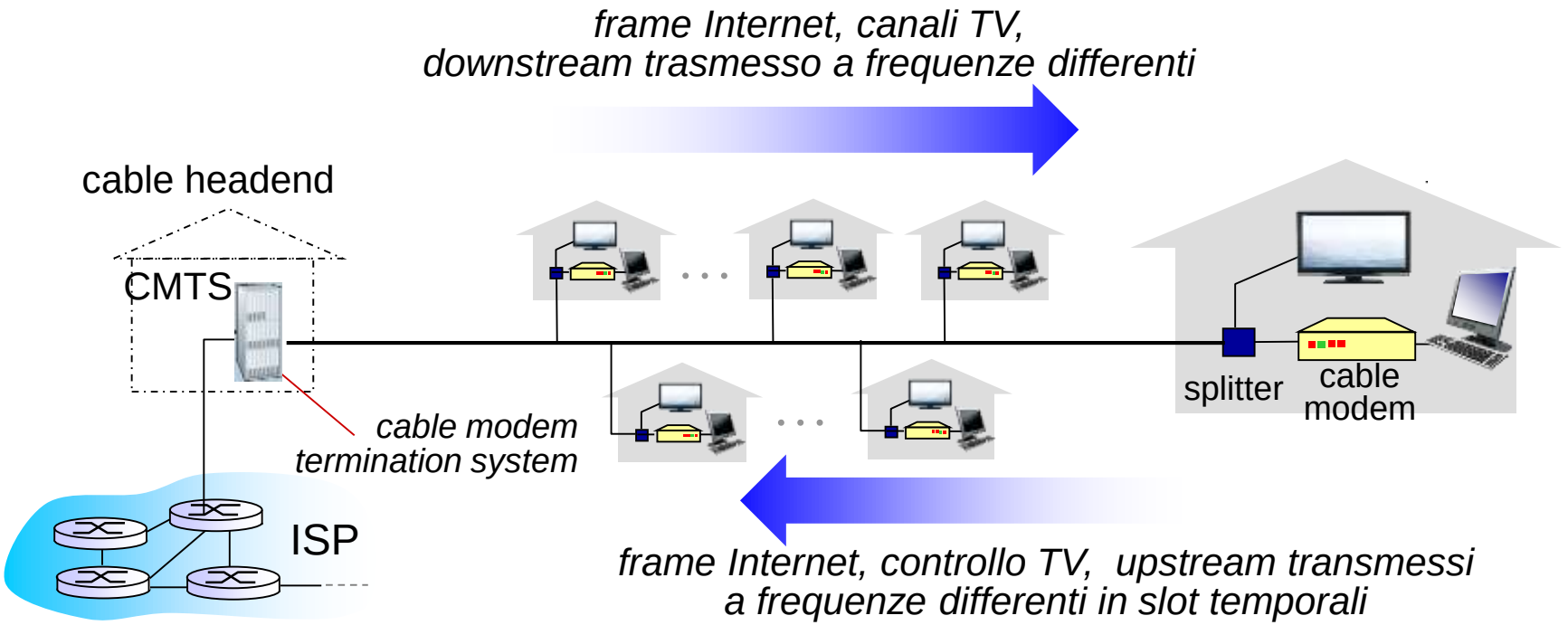
Protocolli "a rotazione"

token passing:

- c'è un *token* di controllo passato da un nodo al successivo sequenzialmente.
- messaggi di token
- difetti:
  - overhead di invio token
  - latenza
  - single point of failure (token)

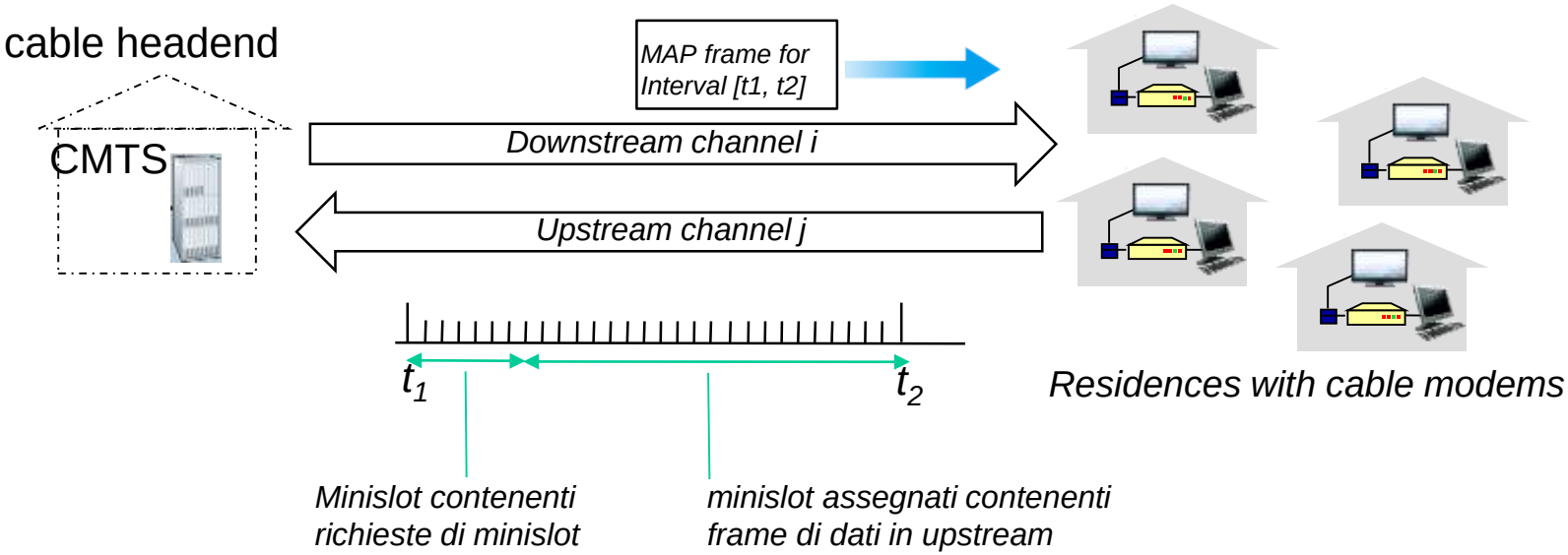


Reti con condivisione di cavo



- *multipli* canali downstream (broadcast) a 40 Mbps
  - un singolo CMTS trasmette nei canali
- *multipli* canali in upstream a 30 Mbps
  - *accesso multiplo: tutti* gli utenti utilizzano gli stessi slot temporali nei canali in upstream (gli altri sono assegnati)

Reti con condivisione di cavo



DOCSIS: data over cable service interface specifications

- FDM nei canali di frequenza in upstream e downstream
- TDM in upstream: alcuni slot assegnati, altri contesi
  - frame MAP in downstream: assegnano gli slot di upstream
  - richieste di slot upstream (e dati) sono trasmette con accesso casuale (binary backoff) negli slot selezionati

Riassunto del protocolli MAC

- *suddivisione di canale,* in tempo, frequenza o codificata
  - Time Division, Frequency Division
- *accesso casuale* (dinamico),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - carrier sensing: facili con alcune tecnologie (cavi), difficile in altri (wireless)
  - CSMA/CD usato in Ethernet
  - CSMA/CA usato in 802.11
- *a rotazione*
  - polling da un nodo centrale, token passing
  - bluetooth, FDDI, token ring

Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

5.6 data center networking

5.7 il cammino di una richiesta web

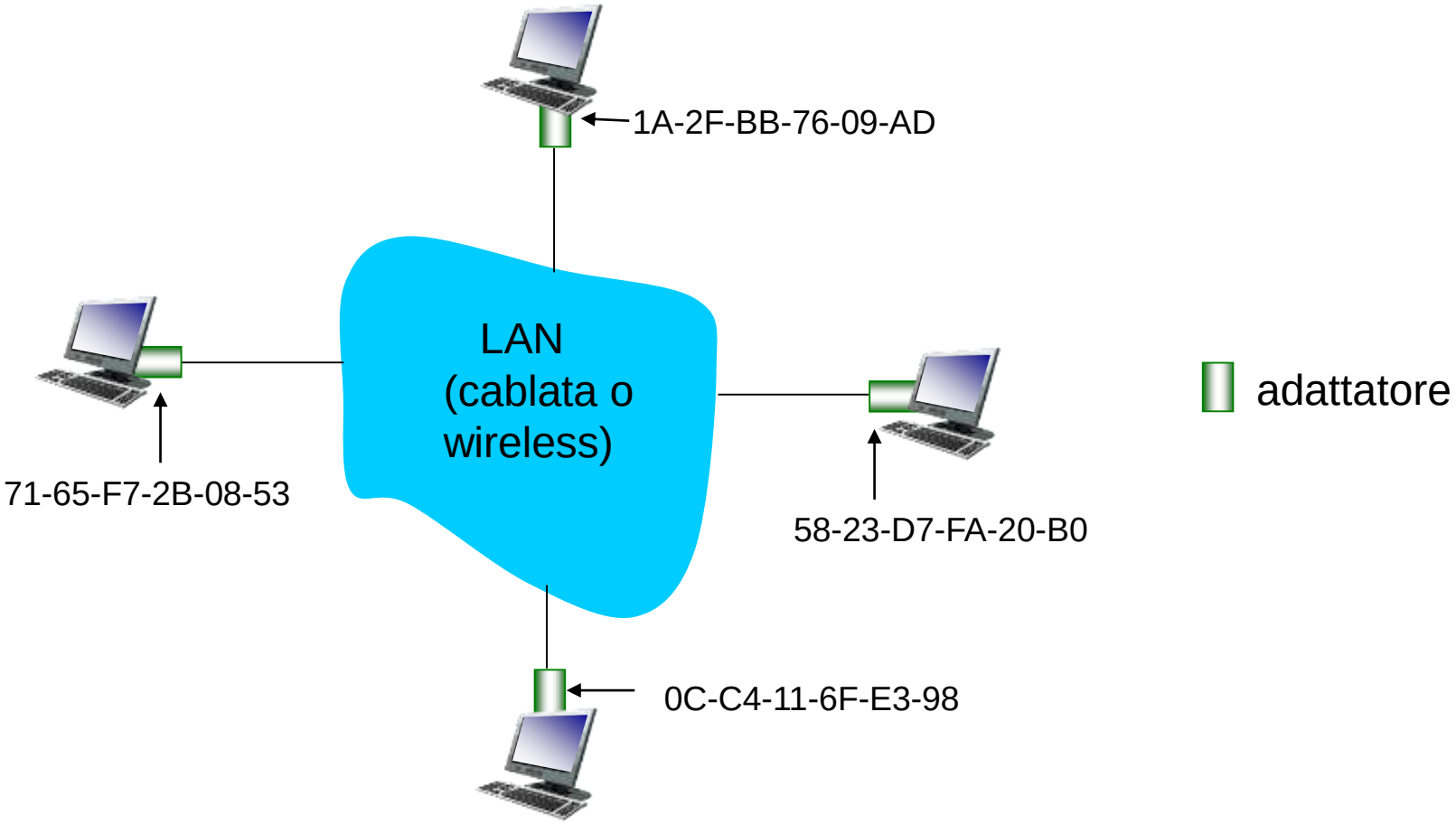
Indirizzi MAC e ARP

- indirizzo IP a 32-bit:
  - indirizzo a *livello di rete* per l'interfaccia
  - usato per il forwarding a layer 3 (network layer)
- indirizzo MAC (o LAN o fisico o Ethernet):
  - funzione: *usato 'localmente" per trasmettere frame da un'interfaccia a un'altra fisicamente connessa (nella stessa rete, dal punto di vista IP)*
  - indirizzo MAC a 48 bit (per la maggior parte delle LAN) impresso nella ROM della NIC, a volte impostabile via software
  - es.: 1A-2F-BB-76-09-AD

notazione esadecimale (base 16)
(ogni "numero" rappresenta 4 bit)

Indirizzi MAC e ARP

ciascun adattatore in una LAN ha un indirizzo *LAN* univoco



Indirizzi MAC (ancora)

- la IEEE sovrintende alla gestione degli indirizzi MAC
- quando una società vuole costruire adattatori, compra un blocco di spazio di indirizzi (unicità degli indirizzi)
- analogia:
  - indirizzo MAC: come il Codice Fiscale
  - indirizzo IP: come l'indirizzo di residenza
- indirizzo MAC piatto ➜ portabilità
  - è possibile spostare una scheda LAN da una LAN a un'altra
- indirizzo IP gerarchico *non* portabile
  - dipende dalla sottorete IP cui il nodo è collegato

ARP: address resolution protocol

Domanda: come si determina l'indirizzo MAC di un'interfaccia se si conosce solo il suo indirizzo IP?

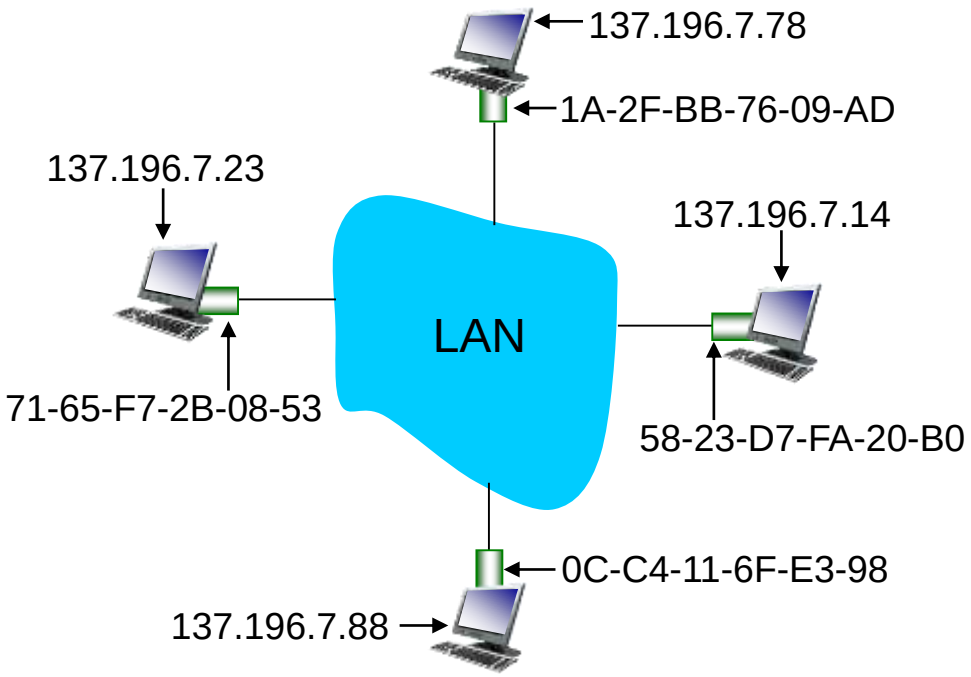


tabella ARP: ogni nodo IP (host, router) nella LAN ne ha una

- mapping indirizzi IP/MAC per i nodi di una LAN:

  < indirizzo IP; indirizzo MAC; TTL>

- TTL (Time To Live): tempo dopo il quale un mapping viene eliminato (tipicamente 20 minuti)

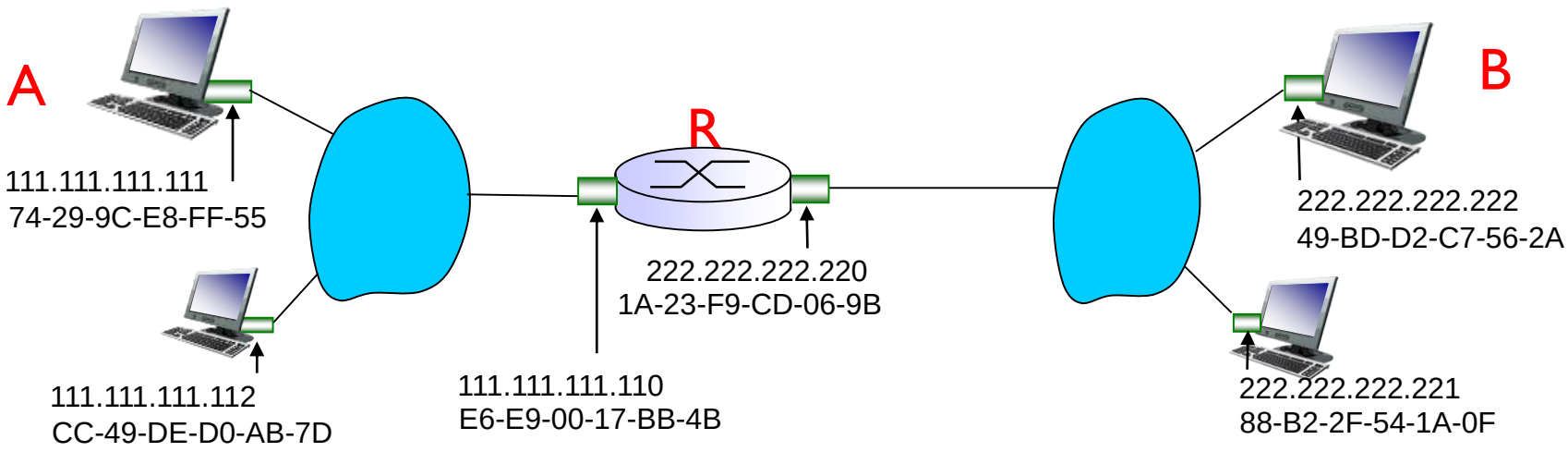
Protocollo ARP: stessa LAN

- A vuole inviare un datagramma a B
  - indirizzo MAC di B non ancora nella tabella ARP di A.
- A invia in broadcasts un pacchetto di richiesta ARP, contenente l'indirizzo IP di B
  - Indirizzo MAC broadcast = FF-FF-FF-FF-FF-FF
  - tutti i nodi nella LAN ricevono la query ARP
- B riceve il pacchetto ARP, risponde ad A con il proprio indirizzo MAC
  - il frame viene inviato verso l'indirizzo MAC di A (unicast)
- A mantiene (cache) la coppia indirizzi IP-to-MAC nella sua ARP table fino a che l'informazione scade (time out)
  - soft state: l'informazione che va in time out (se ne va) a meno di refresh
- ARP è "plug-and-play":
  - i nodi creano la propria tabella ARP *senza l'intervento dell'amministratore del sistema*

Indirizzamento: routing verso un'altra LAN

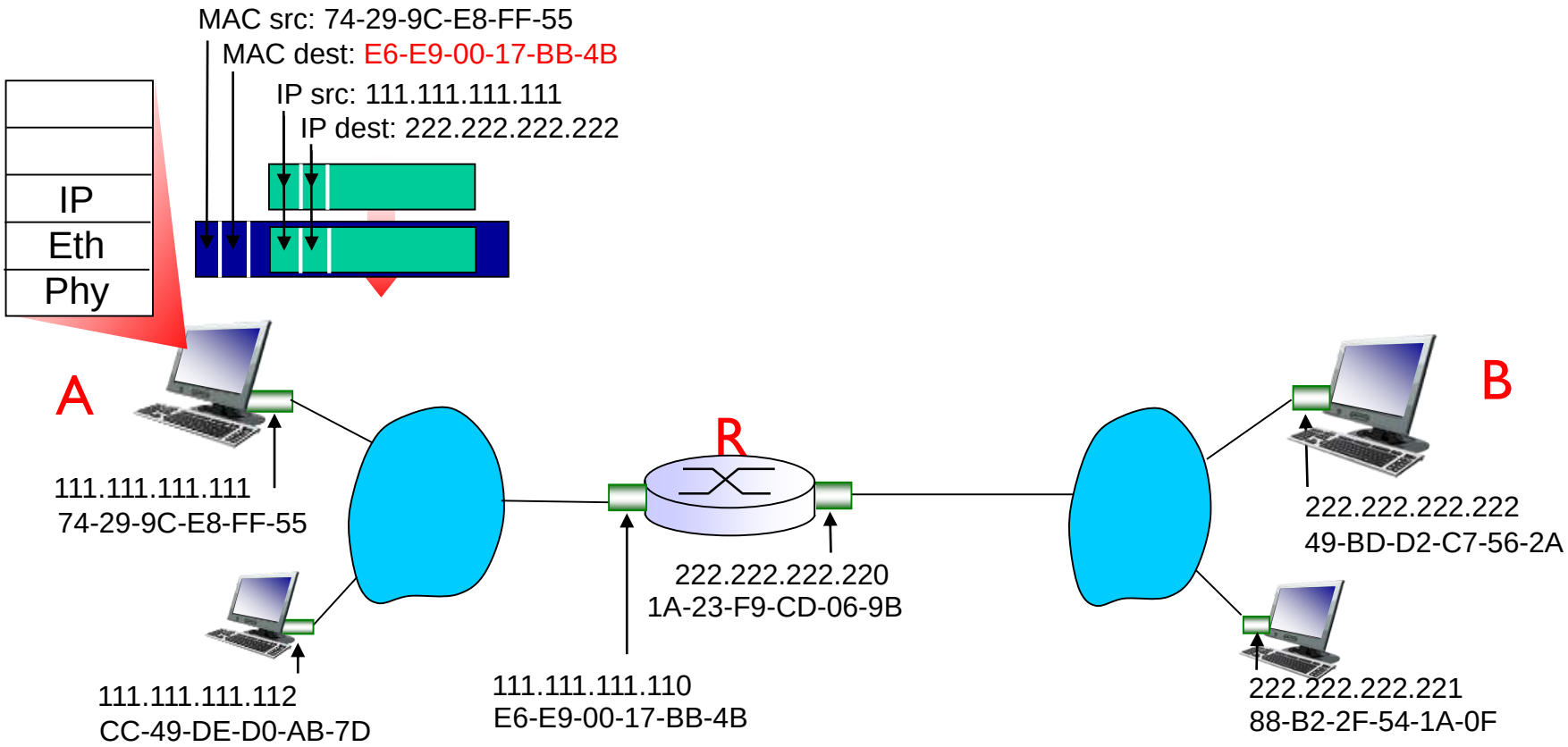
percorso: invio del datagramma da A a B via R

- indirizzamenti – a livello IP (datagrammi) e a livello MAC (frame)
- supponiamo che A sappia l'indirizzo IP di B
- supponiamo che A sappia l'indirizzo IP del router di first hop, R (come?)
- supponiamo che A sappia l'indirizzo MAC di R (come?)



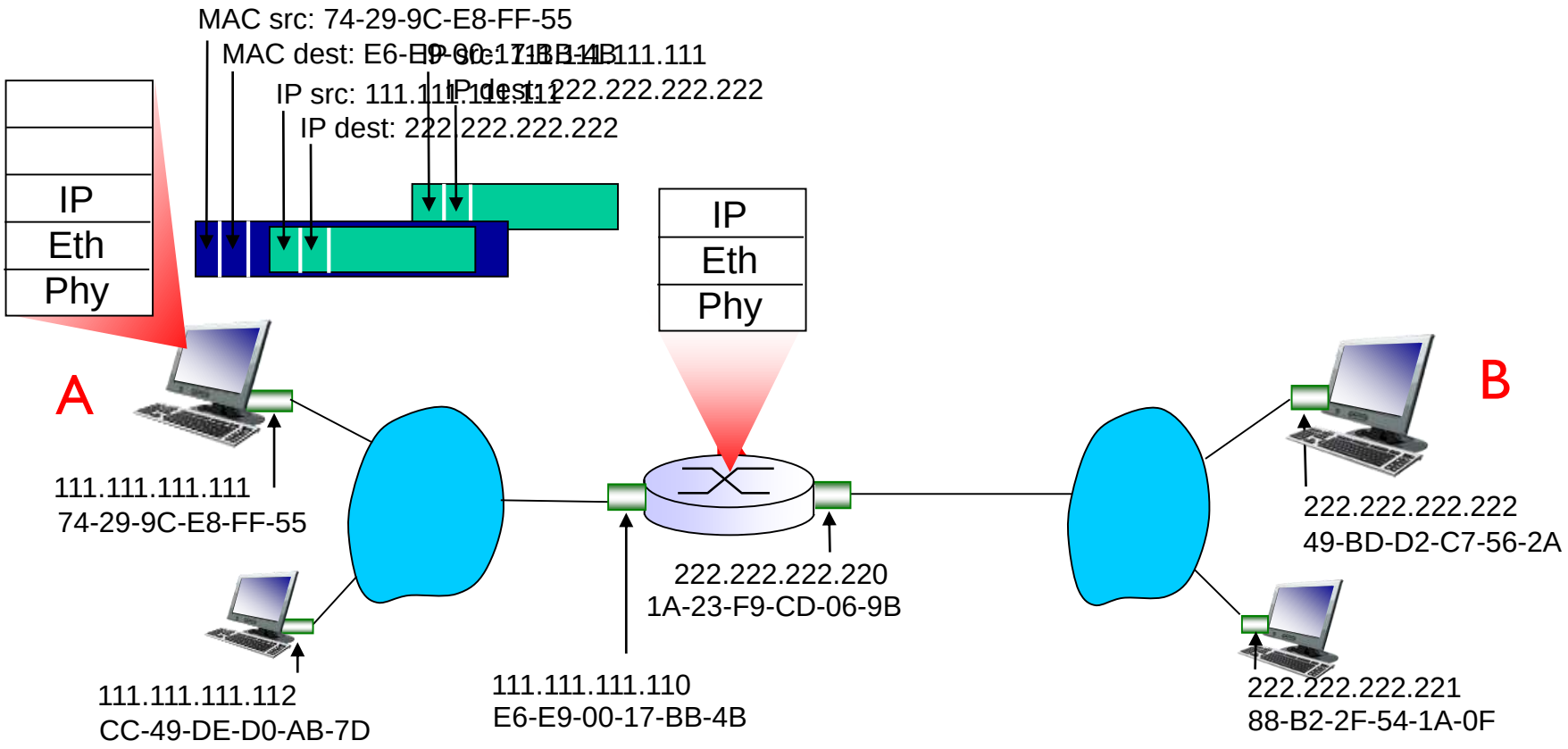
Indirizzamento: routing verso un'altra LAN

- A crea un datagramma IP con origine A e destinazione B
- A crea un frame con l'indirizzo MAC di R come destinazione, il frame contiene il datagramma IP A-verso-B



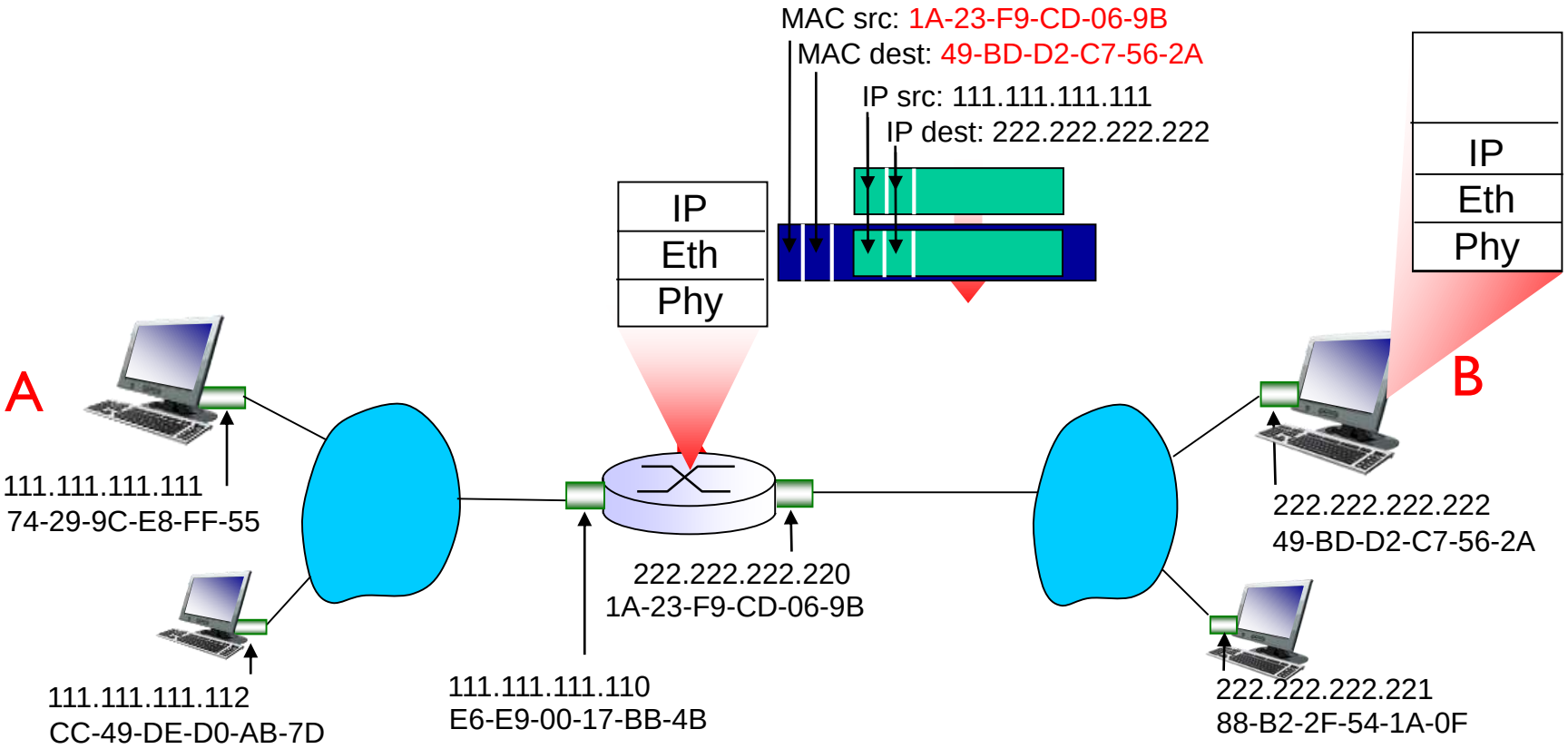
Indirizzamento: routing verso un'altra LAN

- il frame viene inviato da A a R
- il frame viene ricevuto da R, viene estratto il datagramma, e passato al livello IP sovrastante



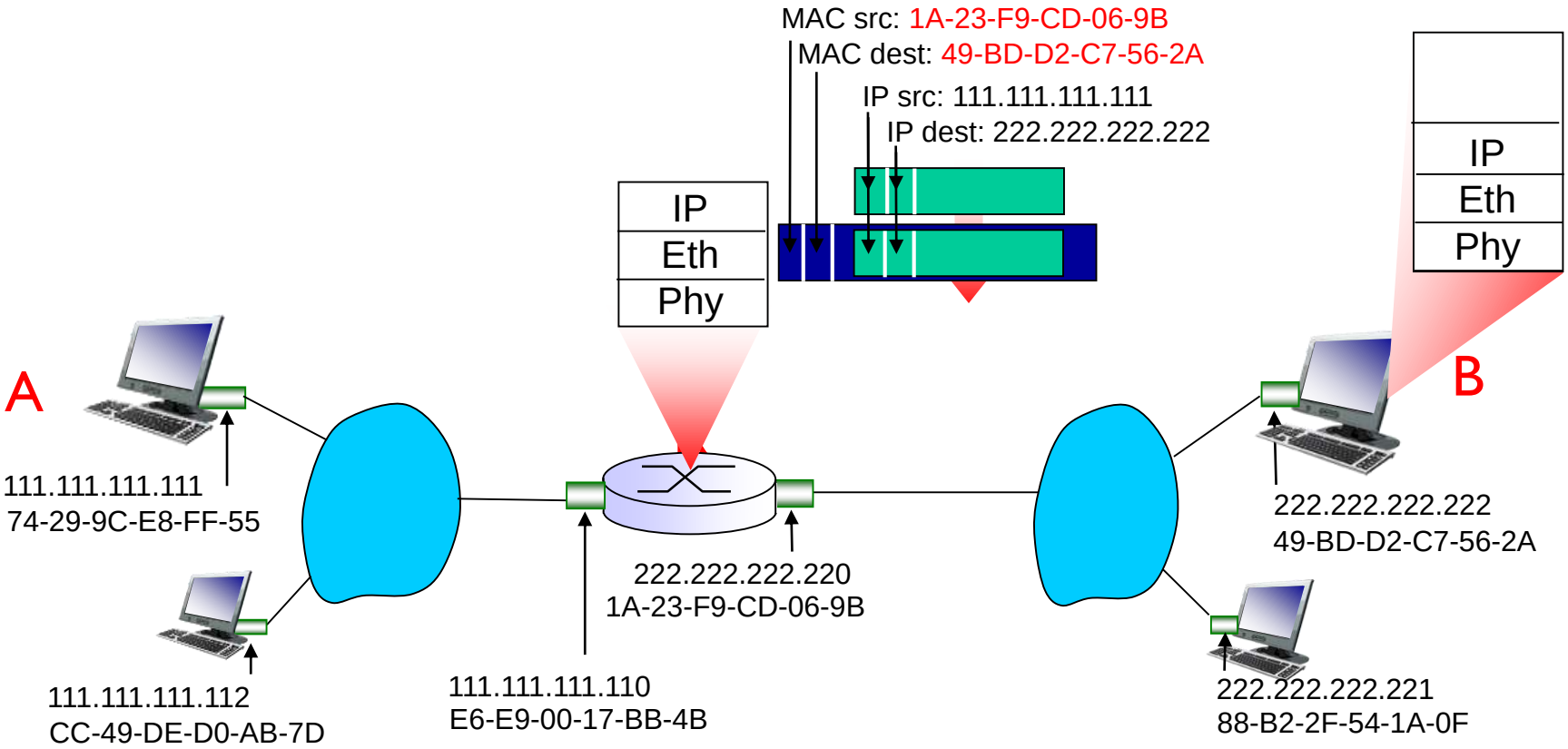
Indirizzamento: routing verso un'altra LAN

- R inoltra il datagramma con sorgente IP A, e destinazione IP B
- R crea il frame con indirizzo MAC di B come destinazione, il frame contiene il datagramma IP A-verso-B



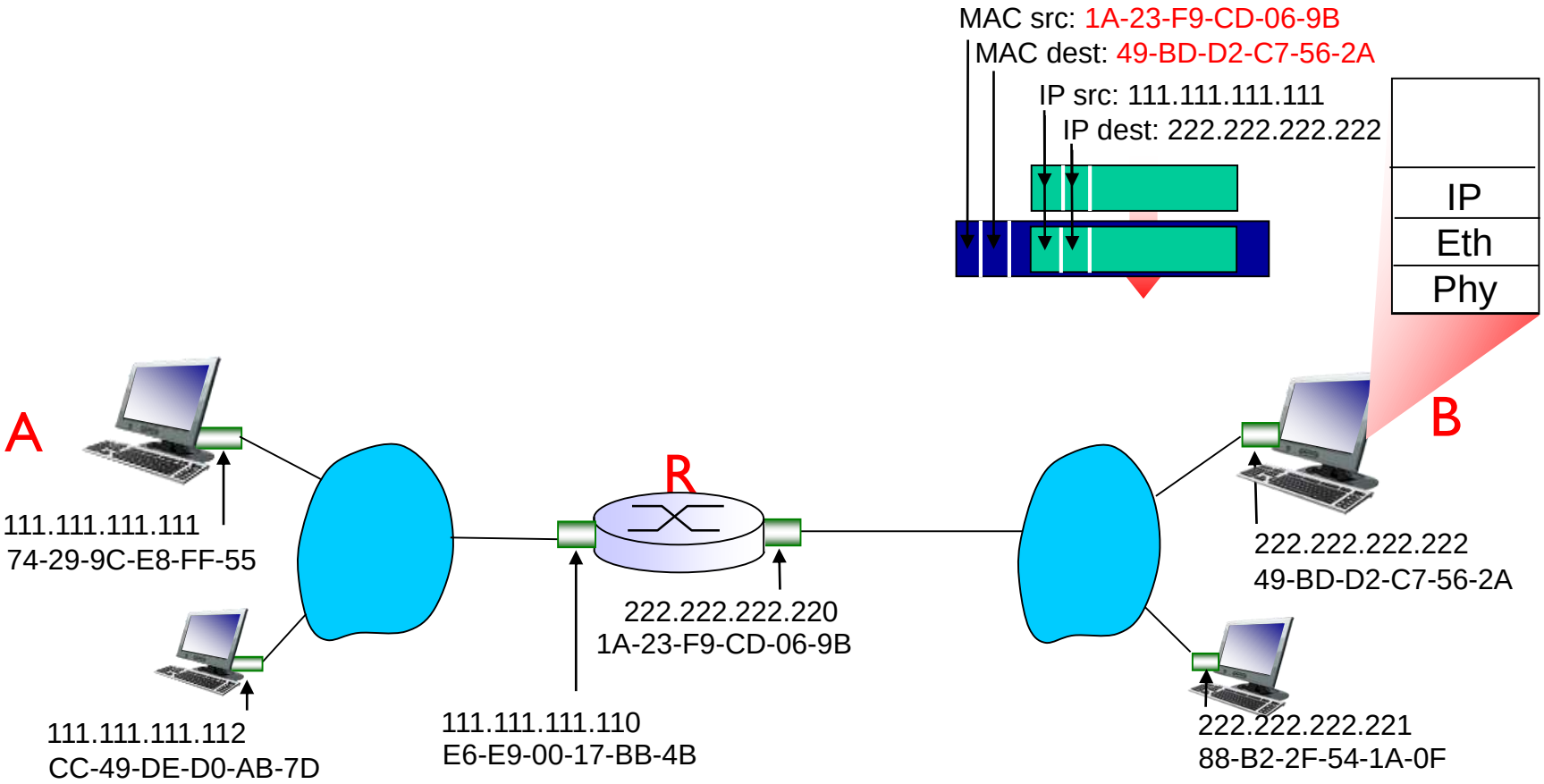
Indirizzamento: routing verso un'altra LAN

- R inoltra il datagramma con sorgente IP A, e destinazione IP B
- R crea il frame con indirizzo MAC di B come destinazione, il frame contiene il datagramma IP A-verso-B



Indirizzamento: routing verso un'altra LAN

- R inoltra il datagramma con sorgente IP A, e destinazione IP B
- R crea il frame con indirizzo MAC di B come destinazione, il frame contiene il datagramma IP A-verso-B



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

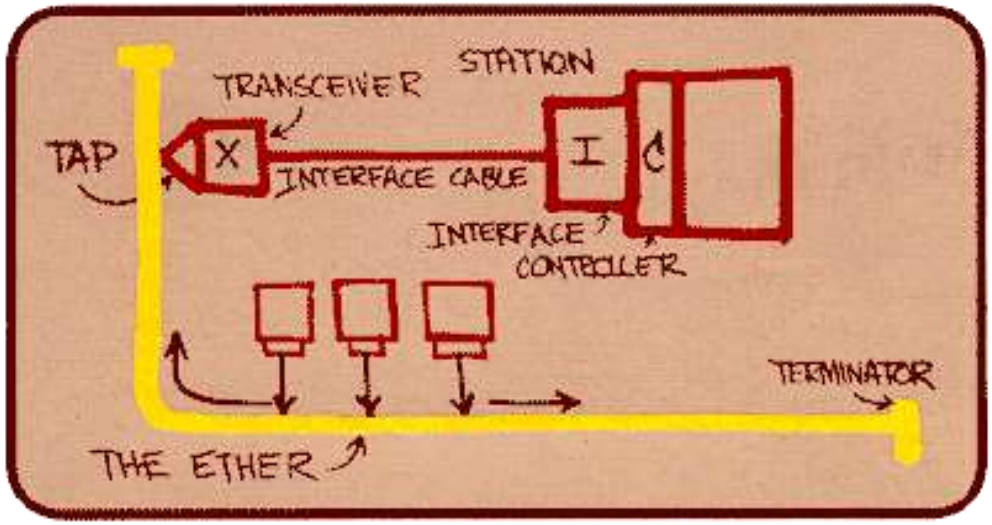
5.6 data center networking

5.7 il cammino di una richiesta web

Ethernet

tecnologia "dominante" per le LAN cablate:

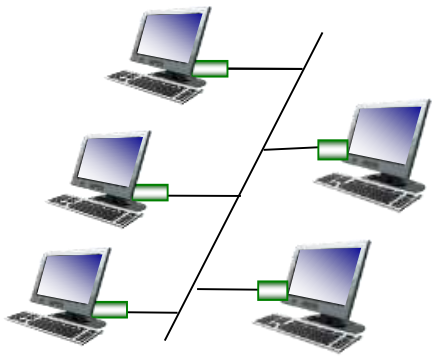
- NIC dal costo limitato
- prima tecnologia LAN con vasta diffusione
- più semplice ed economica di LAN con token e ATM
- al passo con gli aumenti di velocità: 10 Mbps – 10 Gbps



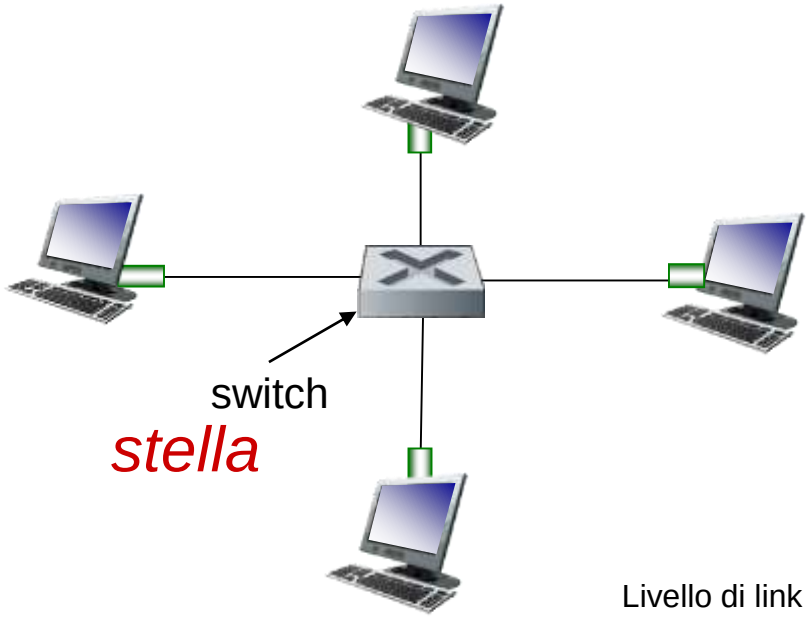
schizzo dell'Ethernet di Metcalfe

Ethernet: topologia fisica

- *bus:* diffusa fino a metà degli anni 90
  - tutti i nodi nello stesso dominio di collisione (possono collidere con tutti gli altri)
- *stella:* prevalgono oggi
  - *switch* attivo al centro
  - ogni "messaggio" viaggia su un (separato) protocollo Ethernet (i nodi non collidono con gli altri)



bus: cavo coassiale



stella

Struttura dei frame Ethernet

l'adattatore trasmittente incapsula i datagrammi IP (o i pacchetti di altri protocolli del livello di rete) in frame Ethernet

preambolo:

- 7 byte dal valore 10101010 seguiti da 1 byte con valore 10101011
- usato per sincronizzare i clock di trasmittente e ricevente

Struttura dei frame Ethernet (altro)

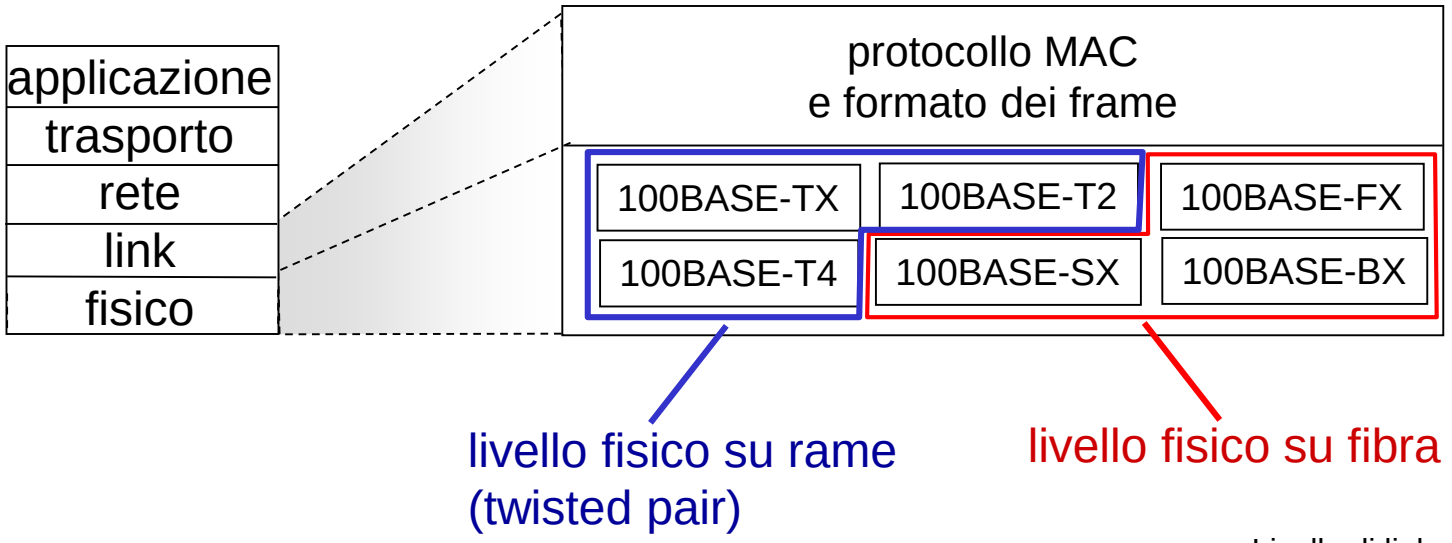
- *indirizzi:* 6 byte per indirizzo MAC di destinazione e 6 per il sorgente
  - se un adattatore riceve un pacchetto contenente come destinazione il proprio indirizzo o l'indirizzo di broadcast (es.: un pacchetto ARP), trasferisce il contenuto del campo dati del pacchetto al livello di rete
  - altrimenti, scarta il frame
- *tipo:* indica il protocollo sovrastante (tipicamente IP ma altri sono possibili, es., Novell IPX, AppleTalk)
- *CRC:* bit di cyclic redundancy check per il ricevente
  - errore rilevato: frame scartato

Ethernet: inaffidabile, senza connessione

- *connectionless:* non c'è handshaking tra NIC sorgente e ricevente
- *inaffidabile:* la NIC ricevente non invia ack o nack alla NIC sorgente
  - i dati dei frame scartati sono recuperati solo da protocolli di livello superiore (es., rdt di TCP), altrimenti sono persi
- protocollo MAC Ethernet: unslotted *CSMA/CD con binary backoff*

Standard Ethernet 802.3: livello di link & fisico

- *molti* standard Ethernet differenti
  - protocolli MAC e formato dei frame comuni
  - differenti velocità: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
  - differenti mezzi del livello fisico: fibra, cavo



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

5.6 data center networking

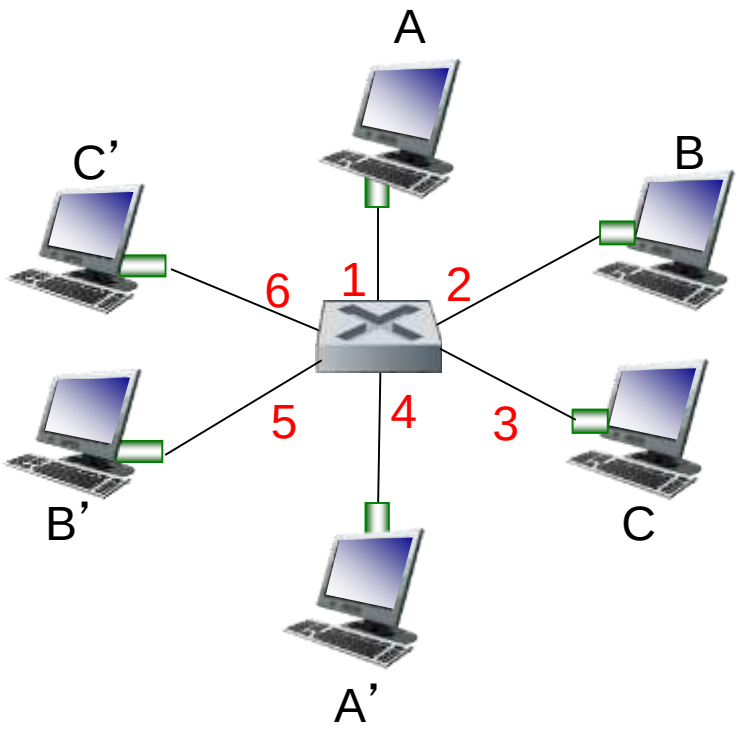
5.7 il cammino di una richiesta web

Switch Ethernet

- dispositivi del link-layer: hanno un ruolo *attivo*
  - effettua store and forward dei frame Ethernet
  - esamina l'indirizzo MAC di destinazione dei frame in arrivo, e li inoltra selettivamente su uno o più link di uscita; quando un pacchetto va inoltrato su un segmento, usa CSMA/CD per accedere al segmento
- *trasparente*
  - gli host non percepiscono la presenza di switch
- *plug-and-play, self-learning*
  - gli switch non hanno bisogno di essere configurati

Switch: trasmissioni *multiple* simultanee

- gli host hanno connessioni dirette e dedicate verso lo switch
- gli switch bufferizzano i pacchetti
- il protocollo Ethernet è usato su *ogni* link entrante, ma non ci sono collisioni; full duplex
  - ogni link costituisce il proprio dominio di collisione
- *switching:* A-verso-A' e B-verso-B' possono trasmettere simultaneamente, senza collisioni

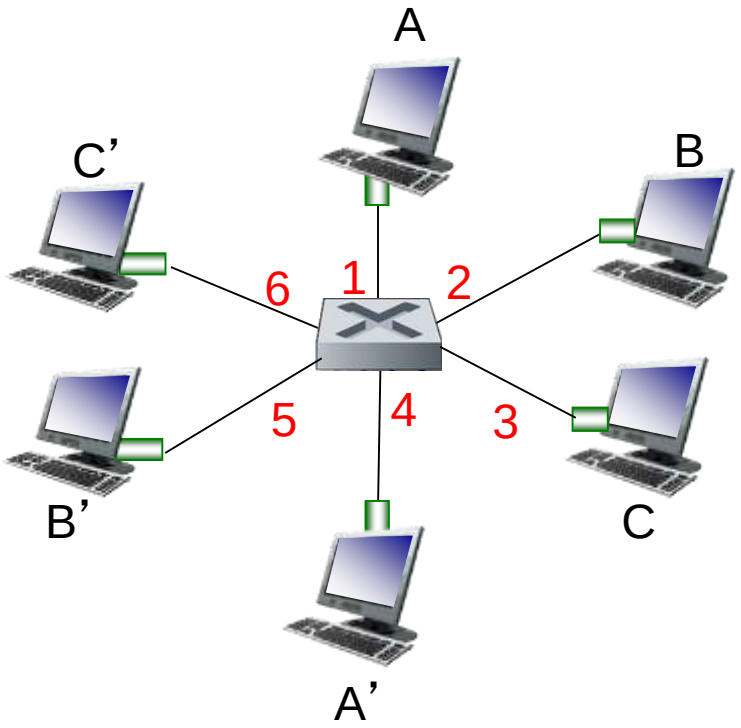


switch con sei interfacce (1,2,3,4,5,6)

Forwarding table negli switch

D: come fa lo switch a sapere che A' è raggiungibile tramite l'interfaccia 4, e B' tramite la 5?

- *R: ogni switch ha una switch table, dove ogni entry è:*
  - *(indirizzo MAC dell'host, interfaccia per raggiungere l'host, time stamp)*
  - *somiglia a una tabella di routing!*



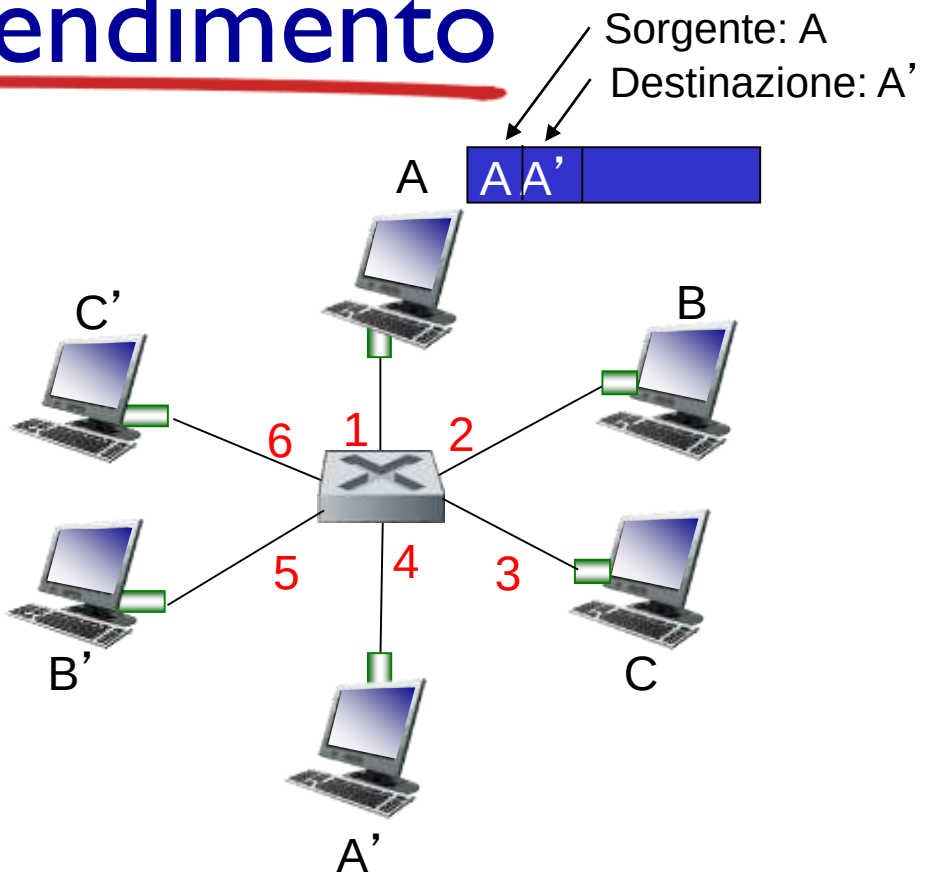
switch con sei interfacce (1,2,3,4,5,6)

D: come sono create le entry, e mantenute nella switch table?

- *con qualcosa simile a un protocollo di routing?*

Switch: auto-apprendimento

- lo switch *apprende* quali nodi possono essere raggiunti attraverso quali interfacce
  - quando viene ricevuto un frame , lo switch "apprende" la collocazione del mittente: il segmento di LAN sull'interfaccia
  - memorizza la coppia mittente/collocazione nella switch table



MAC addr	interface	TTL
A	*1*	*60*

switch table (inizialmente vuota)

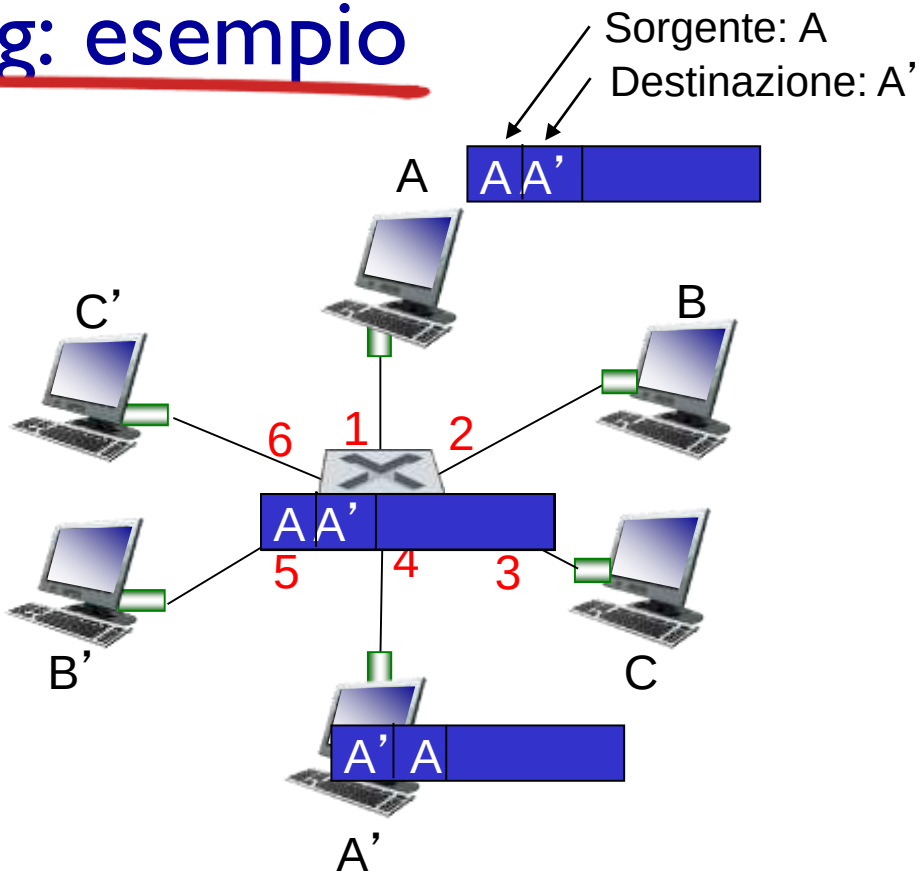
Switch: filtraggio e inoltro dei frame

quando uno switch riceve un frame:

1. memorizza il link in ingresso, e il MAC address dell'host mittente
2. ricerca nella switch table l'indirizzo MAC di destinazione
3. if trova la entry corrispondente alla destinazione
   then {
   if la destinazione è sul segmento dal quale arriva il frame
   then scarta il frame
   else inoltra il frame sull'interfaccia indicata dalla entry
   }
   else flood /* inoltra su tutte le interfacce tranne quella di arrivo */

Self-learning, forwarding: esempio

- destinazione del frame, A', posizione ignota: *flood*
- destinazione A posizione nota: *invio selettivo su un solo link*

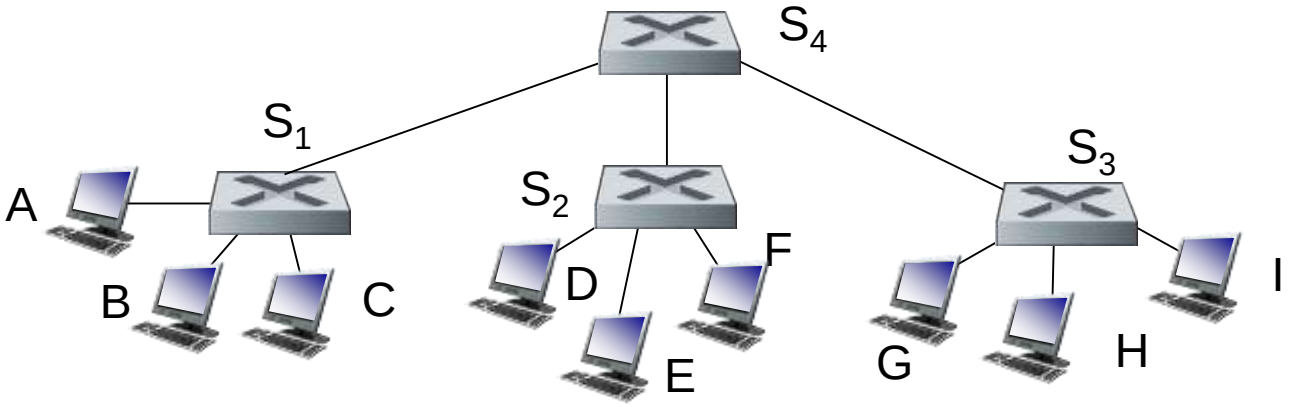


MAC addr	interface	TTL
A	*1*	*60*
A'	*4*	*60*

switch table (inizialmente vuota)

Connettere più switch

- gli switch possono essere connessi tra loro

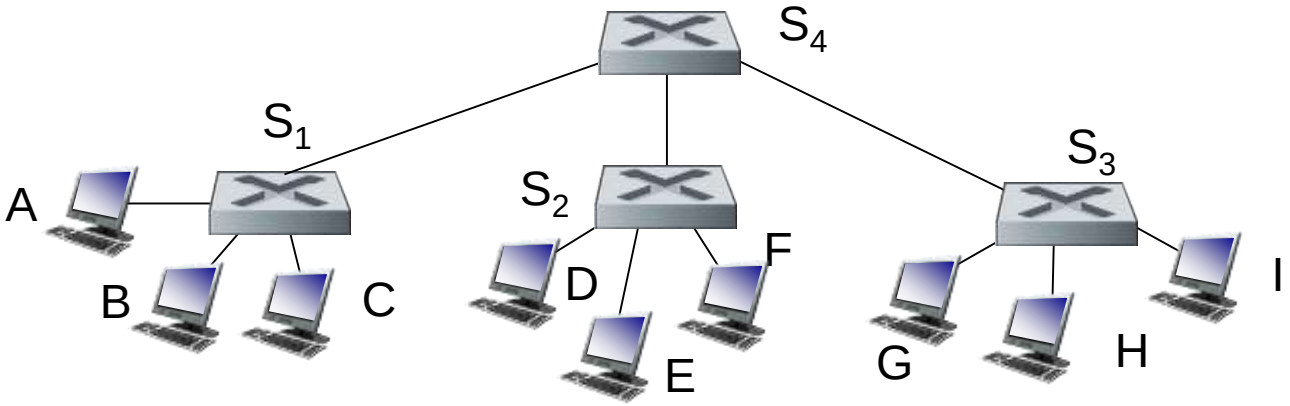


D: nell'invio da A a G – come fa' S_1 a inoltrare il frame destinato a G via S_4 e poi S_3?

- *R:* self learning! (funzione *esattamente* allo stesso modo dello switch singolo!)

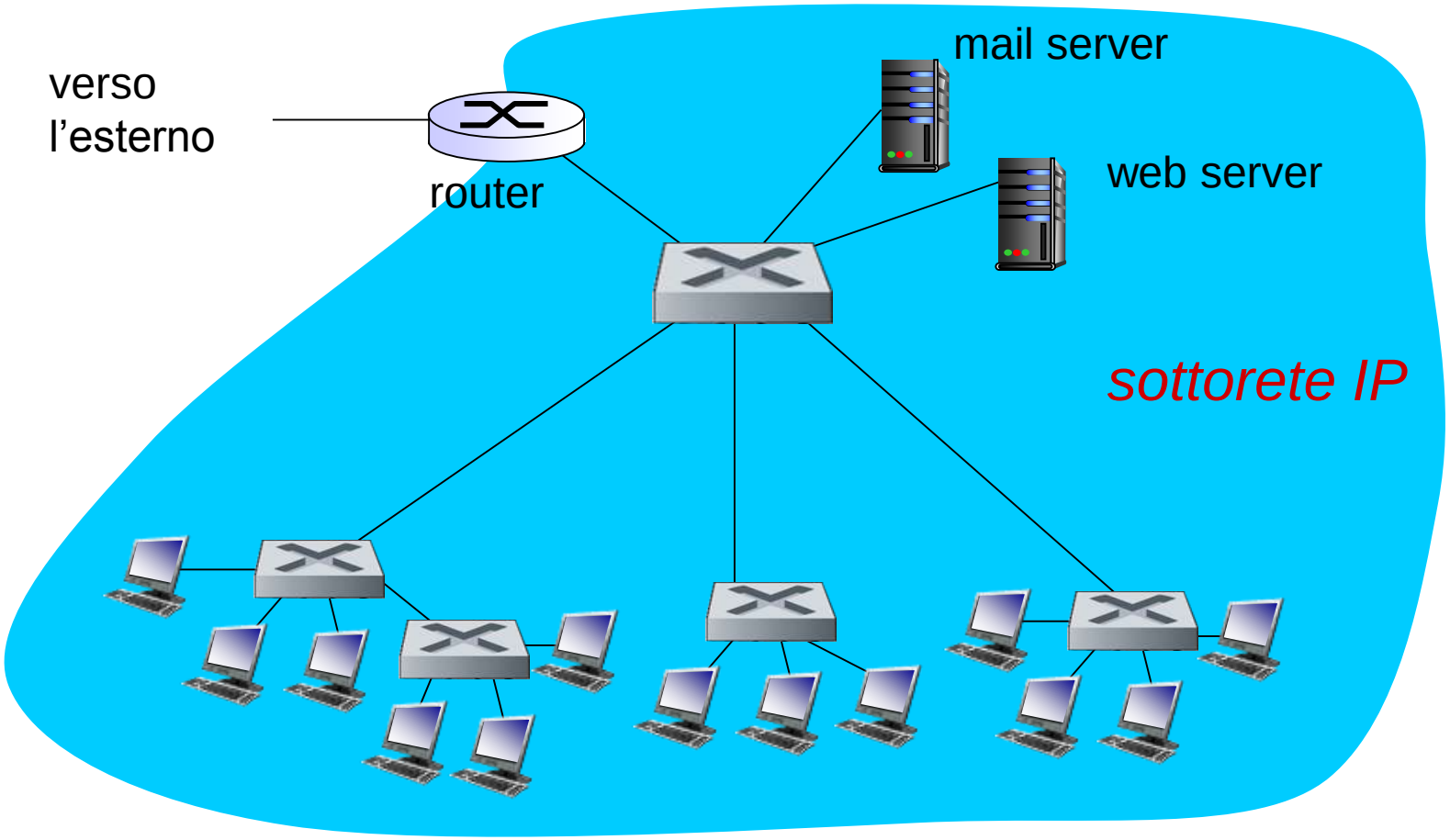
Esempio di self-learning con più switch

Supponiamo che C invii frame a I, e I risponda a C



- *D:* quali switch table si formeranno in S_1, S_2, S_3, S_4?

Esempio di rete istituzionale



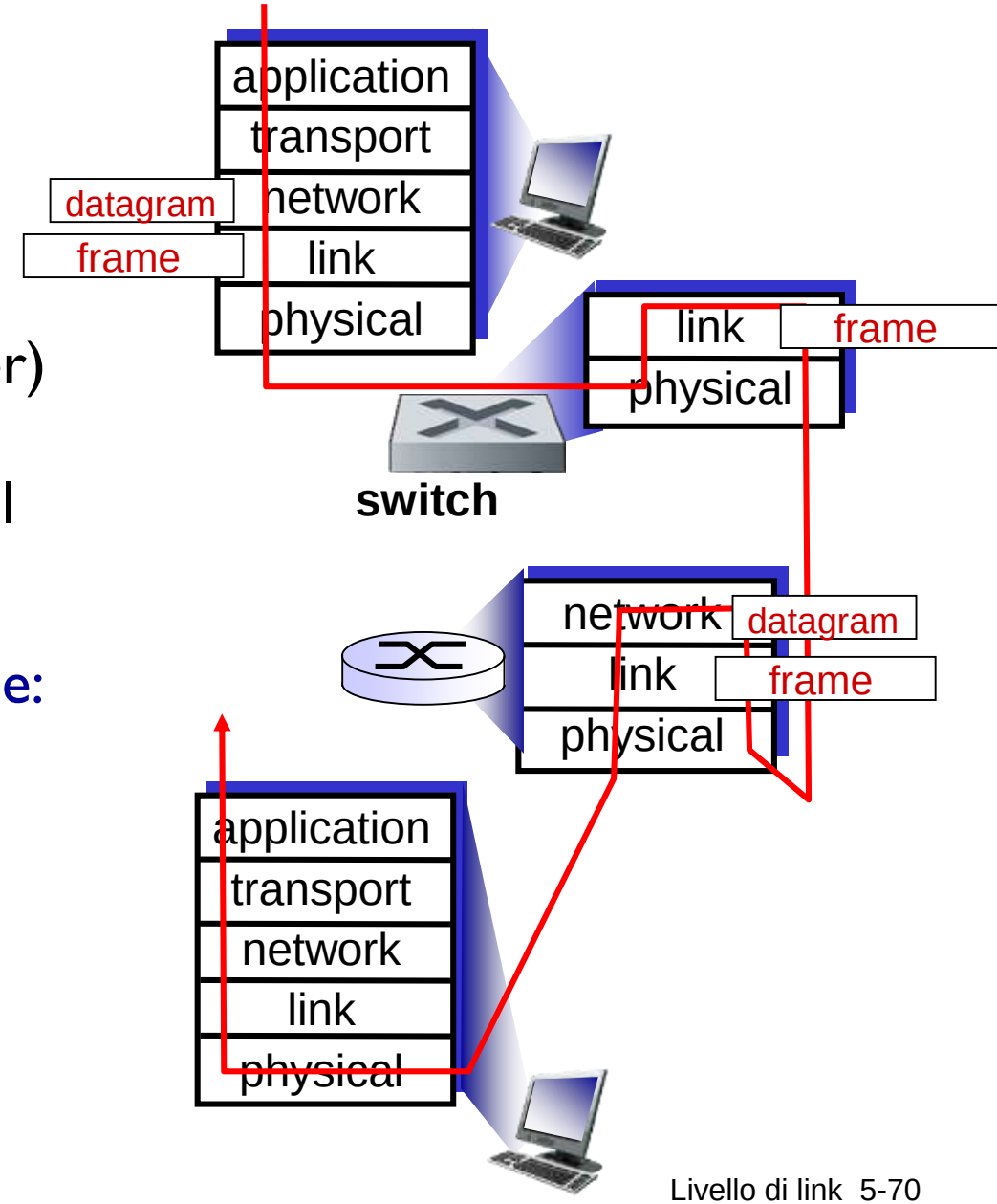
Switch vs. router

entrambi store-and-forward:

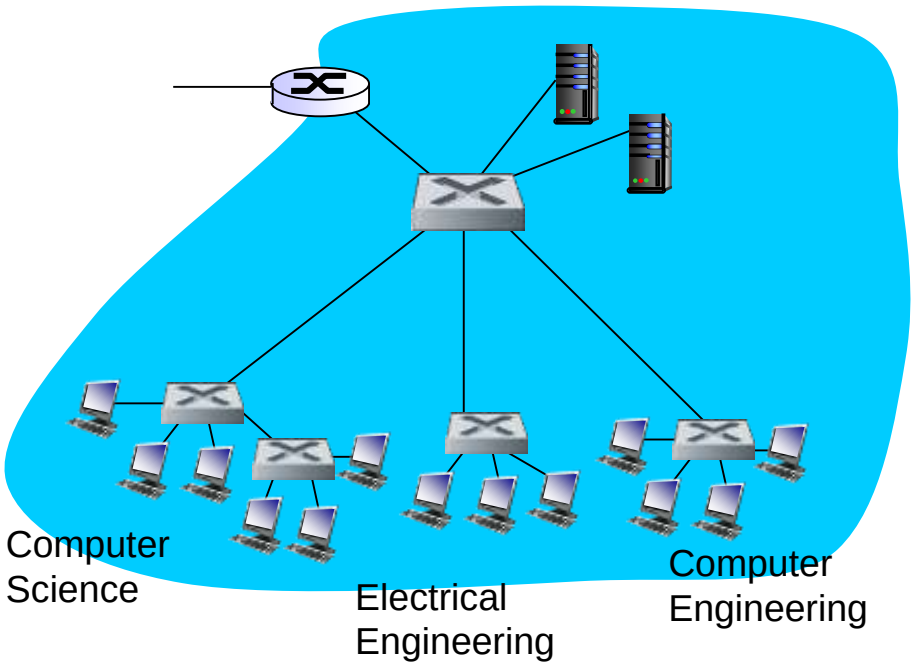
- *router:* dispositivi network-layer (esaminano le intestazioni del network-layer)
- *switch*: dispositivi link-layer (esaminano le intestazioni del link-layer)

entrambi hanno forwarding table:

- *router:* elaborano le tabelle usando algoritmi di routing e indirizzi IP
- *switch:* apprendono le forwarding table usando flooding, learning e indirizzi MAC



VLAN: motivazioni



scenario:

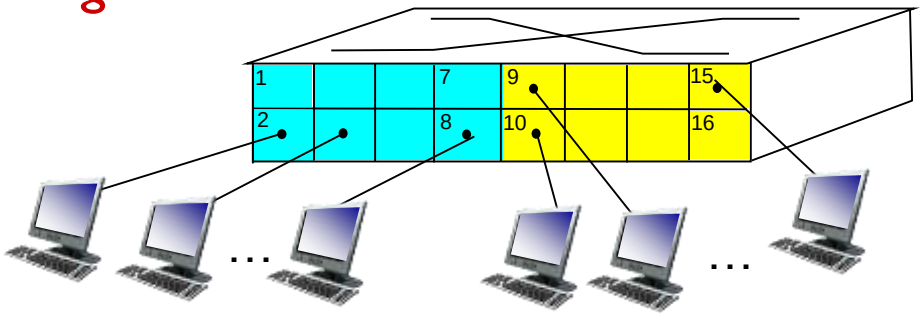
- alcuni utenti di CS spostano l'ufficio in EE, ma vogliono comunicare con gli altri utenti di CS
- singolo dominio di broadcast:
  - tutto il traffico broadcast a layer-2 (ARP, DHCP, indirizzi MAC di destinazione sconosciuti) devono attraversare l'intera LAN
  - problemi di sicurezza/privacy, efficienza

VLAN

Virtual Local Area Network

gli switch che supportano le VLAN possono essere configurati per definire LAN ***virtuali*** multiple su una singola infrastruttura fisica della LAN.

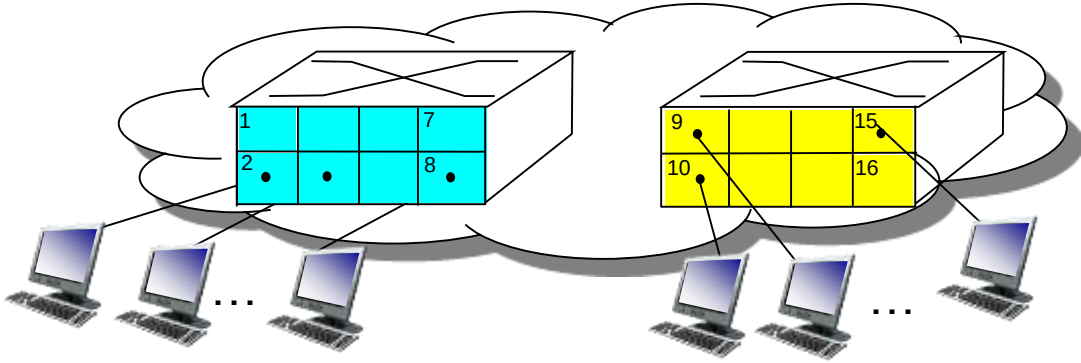
VLAN port-based: le porte degli switch sono raggruppate (dal software di gestione degli switch) in modo che il *singolo* switch fisico ……



Electrical Engineering (porte VLAN 1-8)

Computer Science (porte VLAN 9-15)

… operi come switch virtuali *multipli*

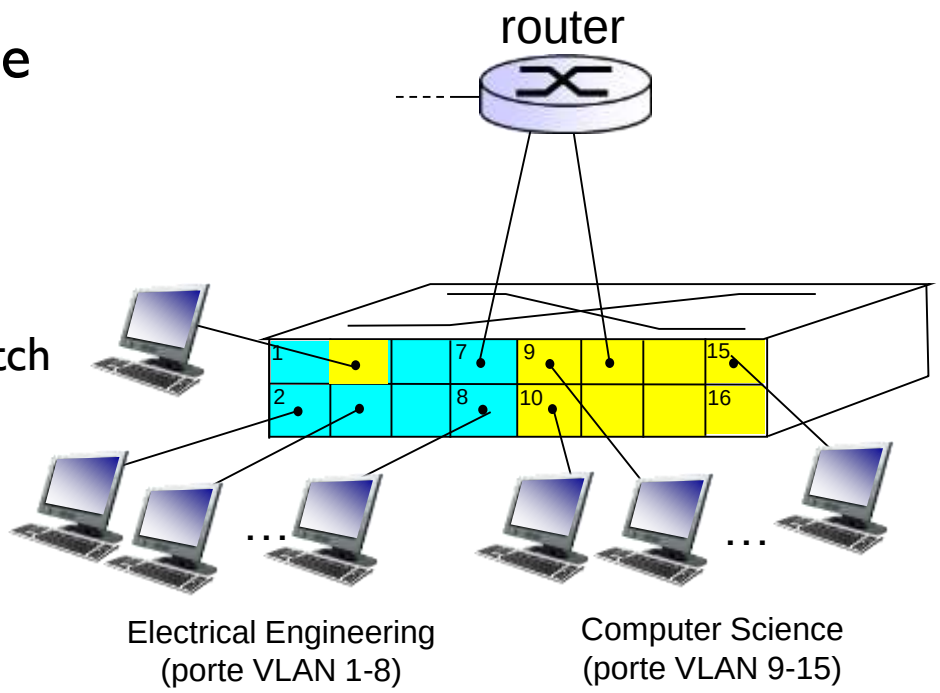


Electrical Engineering (VLAN ports 1-8)

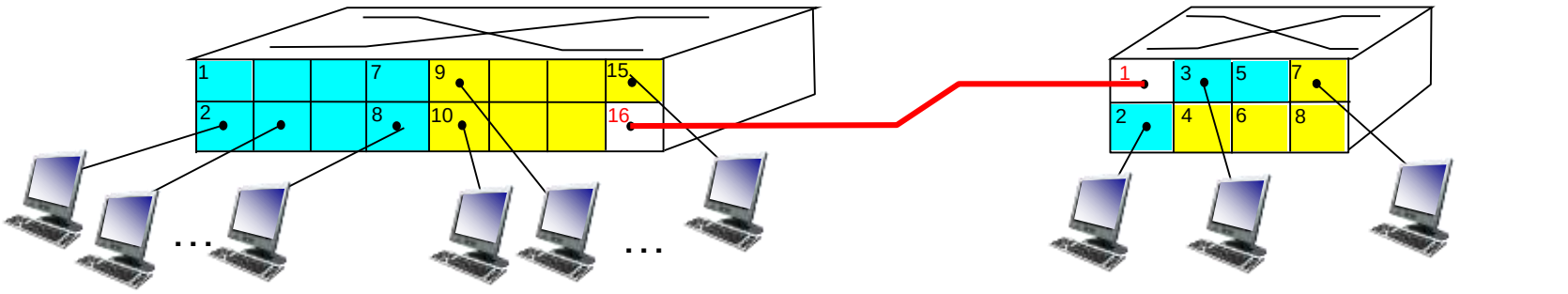
Computer Science (VLAN ports 9-16)

VLAN port-based

- *isolamento del traffico:* un frame da/verso le porte 1-8 può raggiungere *solo* le porte 1-8
  - si possono definire VLAN basate sugli indirizzi MAC degli endpoint, piuttosto che sulle porte dello switch
- *appartenenza dinamica:* le porte possono essere assegnate dinamicamente alle VLAN
- *forwarding tra VLAN:* effettuato tramite routing (come per switch separati)
  - adesso vengono prodotti switch-router combinati



VLAN definite su switch multipli



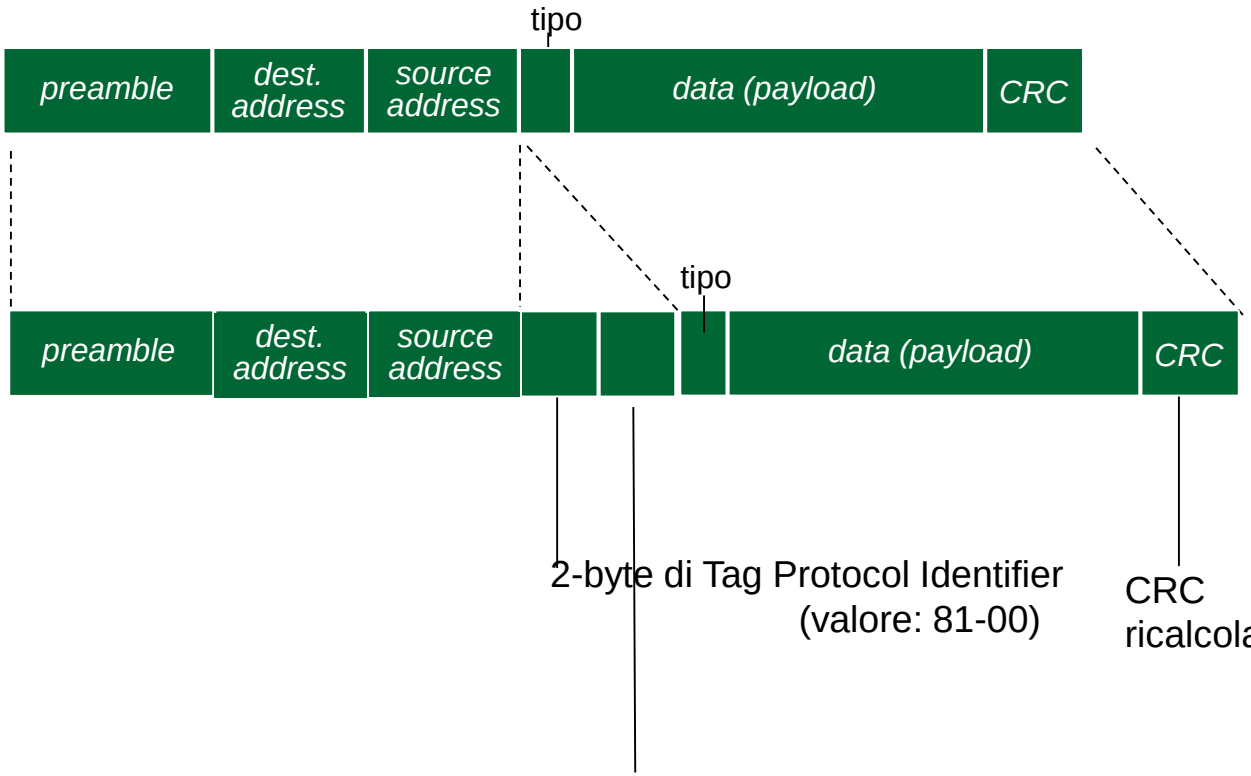
Electrical Engineering
(porte VLAN 1-8)

Computer Science
(porte VLAN 9-15)

Porte 2,3,5 appartengono alla VLAN EE
Porte 4,6,7,8 appartengono alla VLAN CS

- *porta trunk:* trasporta i frame tra VLAN definite su switch fisici multipli
  - i frame inoltrati tra gli switch non possono appartenere a una specifica VLAN (devono trasportare informazioni sulla VLAN ID)
  - il protocollo 802.1q aggiunge/rimuove intestazioni addizionali per i frame inoltrati tra le porte trunk

Formato di un frame VLAN (802.1Q)



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

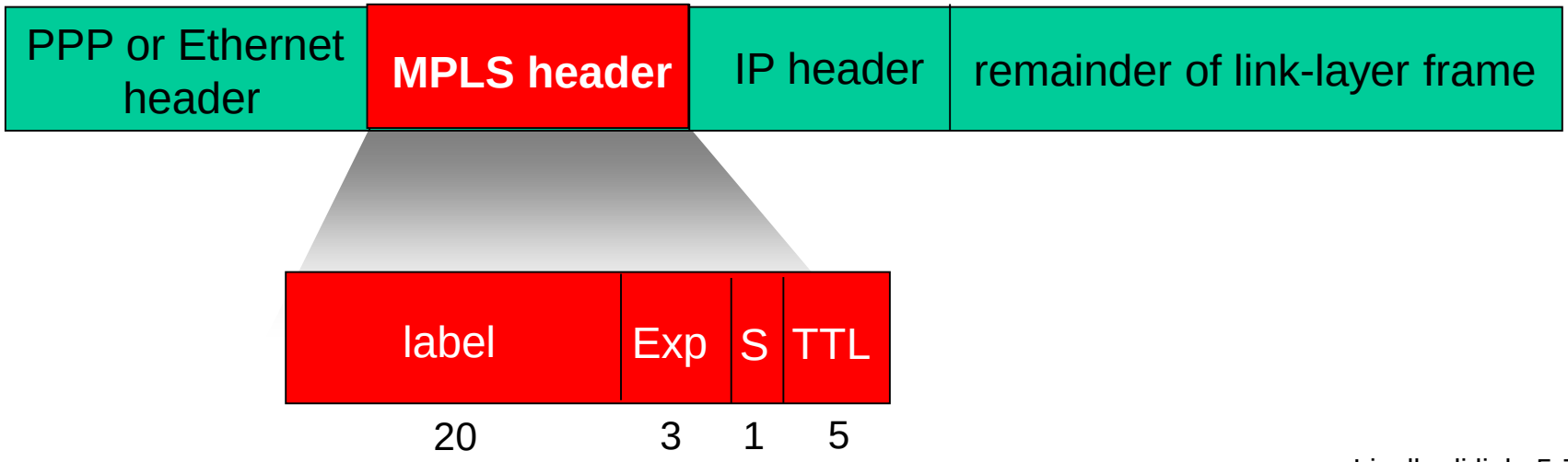
5.5 virtualizzazione di un link: MPLS

5.6 data center networking

5.7 il cammino di una richiesta web

Multiprotocol label switching (MPLS)

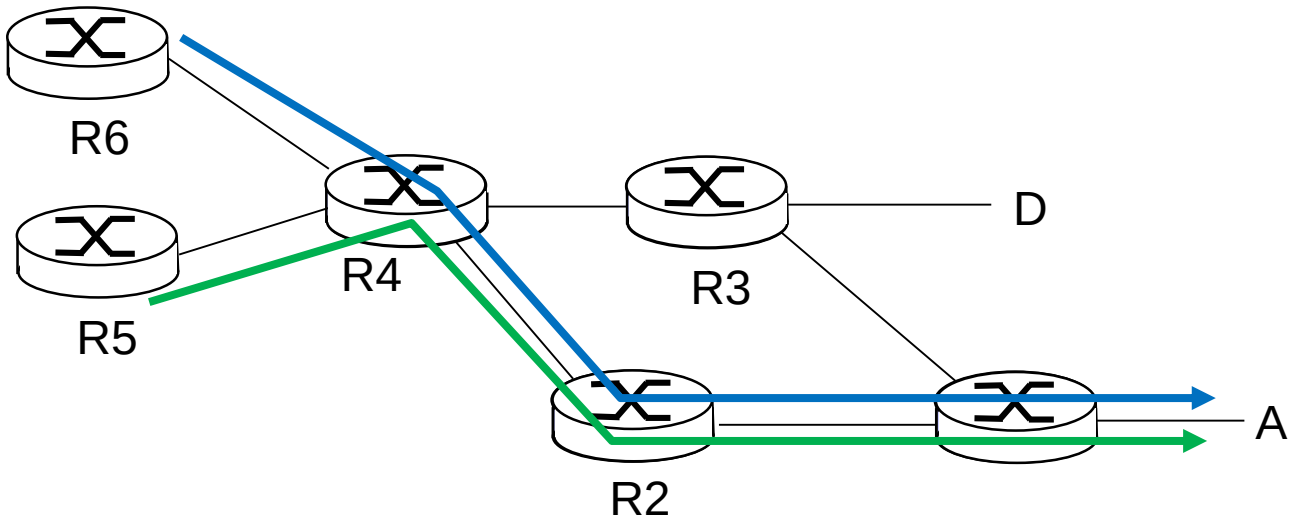
- obiettivo iniziale: velocizzare l'inoltro IP usando un'etichetta di lunghezza stabilita (invece degli indirizzi di destinazione IP)
  - lookup più veloce usando identificativi a lunghezza fissa (piuttosto che prefix matching)
  - l'idea viene presa in prestito dall'approccio del VC
  - ma i datagrammi IP mantengono ancora l'indirizzo IP!



Router MPLS

- detti anche label-switched router (router a commutazione di etichetta)
- inoltrano i pacchetti sull'interfaccia di uscita analizzando l'etichetta MPLS (*non ispezionano l'indirizzo IP*)
  - forwarding table MPLS diversa dalle forwarding table IP
- *flessibilità:* decisioni di forwarding MPLS possono *differire* da quelle dell'IP
  - usano gli indirizzi destinazione e sorgente per instradare flussi verso la stessa destinazione su percorsi differenti ('ingegneria del traffico')
  - cambiano rotte velocemente se un link cade: cammini di backup pre-calcolati (utile per il VoIP)

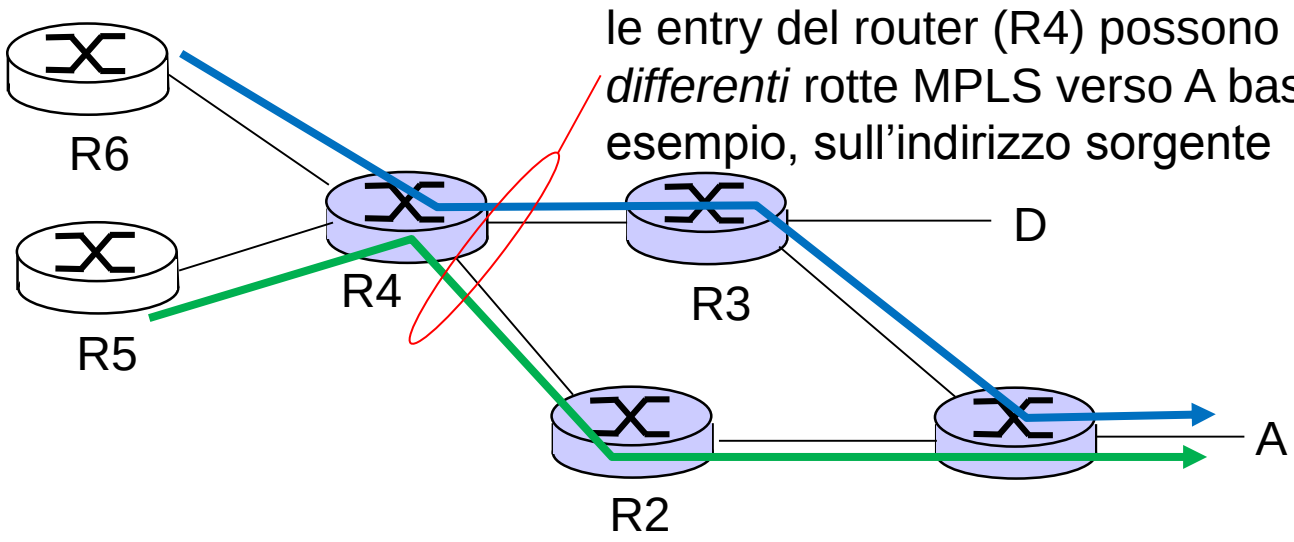
Confronto cammini MPLS e IP



- *routing IP: cammino verso la destinazione determinato solo dall'indirizzo di destinazione*

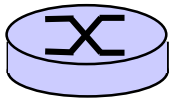

router IP

Confronto cammini MPLS e IP



- *routing IP:* *cammino verso la destinazione determinato solo dall'indirizzo di destinazione*
- *routing MPLS:* il cammino verso la destinazione si può basare sia sull'indirizzo mittente che di destinazione
  - *fast reroute:* precalcola rotte di backup

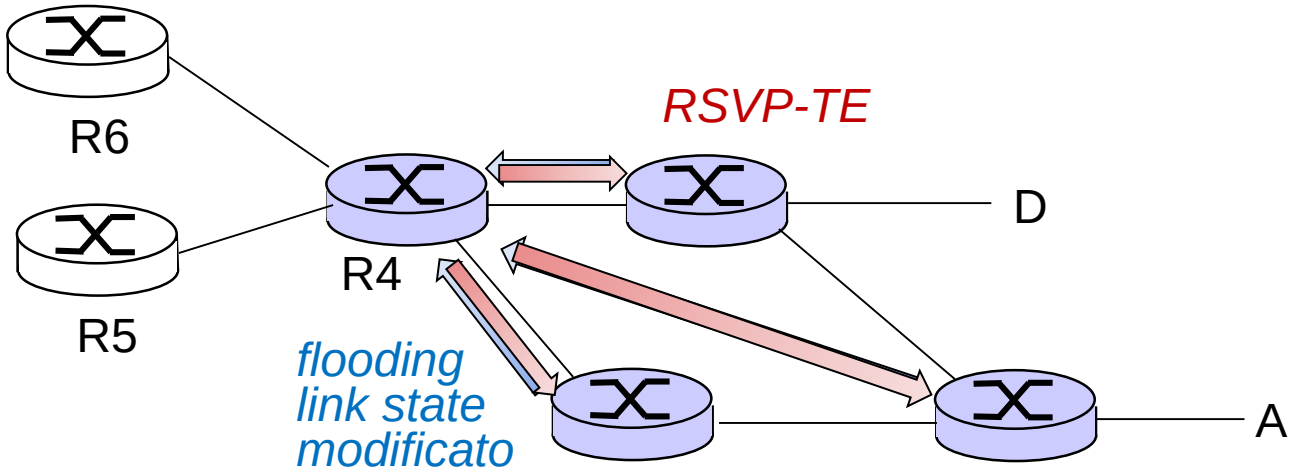
router solo-IP



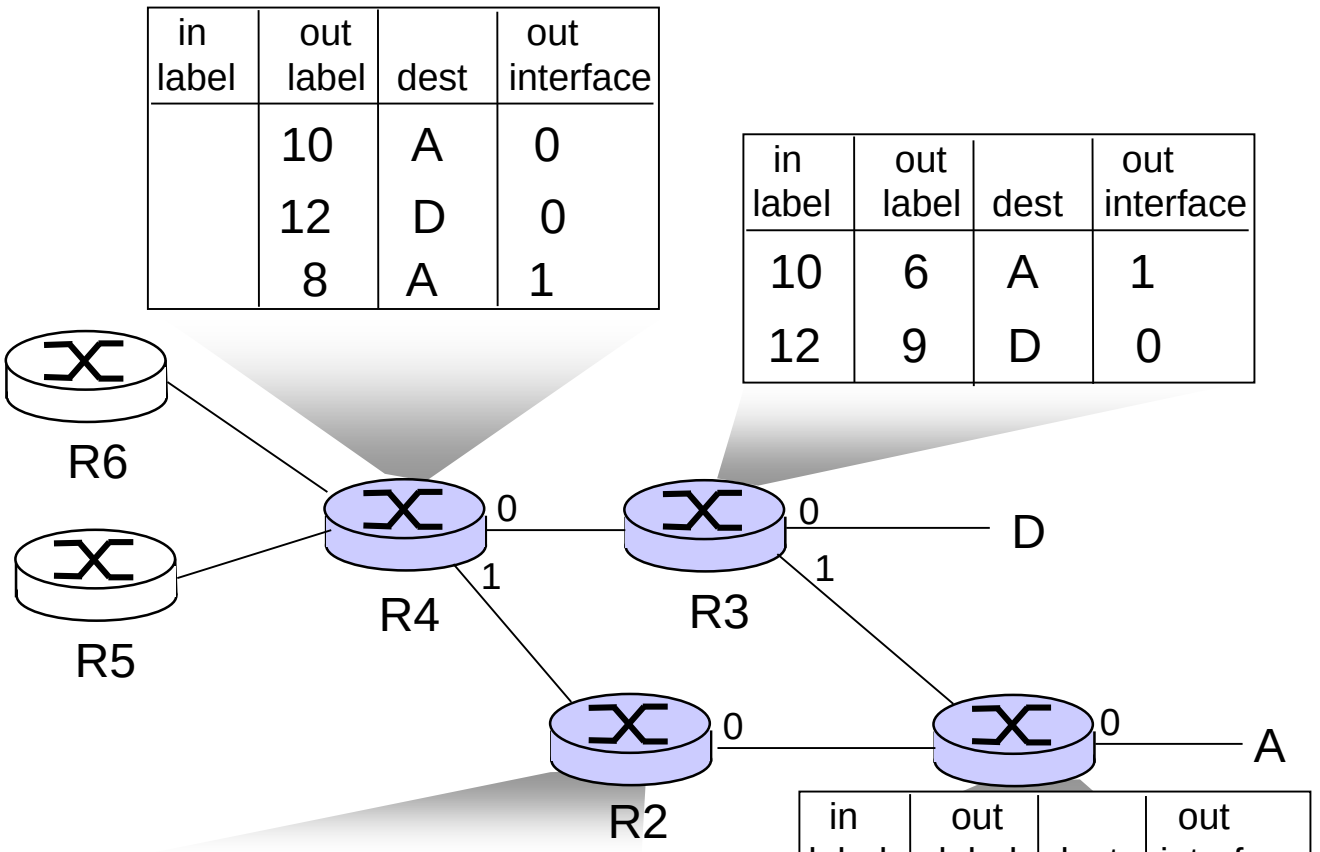
router MPLS e IP

Segnali MPLS

- modificano i protocolli di flooding dei link-state OSPF, IS-IS per trasportare informazioni usate dal routing MPLS,
  - es., bandwidth del link, quantità di bandwidth "riservata"
- *le entry dei router MPLS usano il protocollo dei segnali RSVP-TE per impostare il forwarding MPLS verso i router downstream*



Forwarding table MPLS



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

5.6 data center networking

5.7 il cammino di una richiesta web

Reti di data center

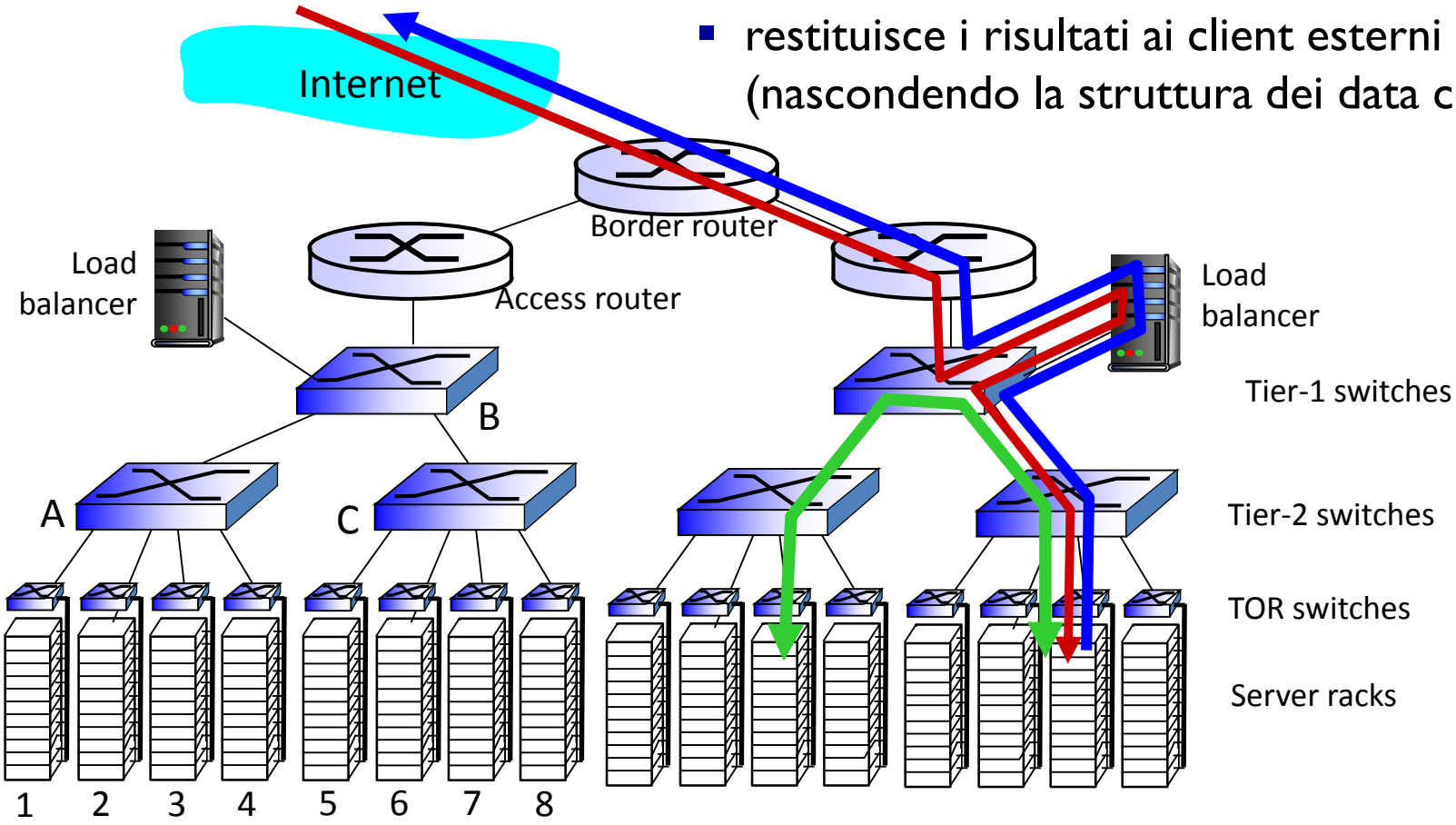
- decine-centinaia di migliaia di host, spesso fortemente accoppiati e fisicamente vicini:
  - e-business (es. Amazon)
  - content-server (es., YouTube, Akamai, Apple, Microsoft)
  - motori di ricerca, data mining (es., Google)
- sfide:
  - applicazioni molteplici, ognuna al servizio di un gran numero di client
  - gestione/bilanciamento del carico, per evitare bottleneck nel carico, nell'elaborazione e nel traffico

Container Microsoft,
data center di Chicago

Reti di data center

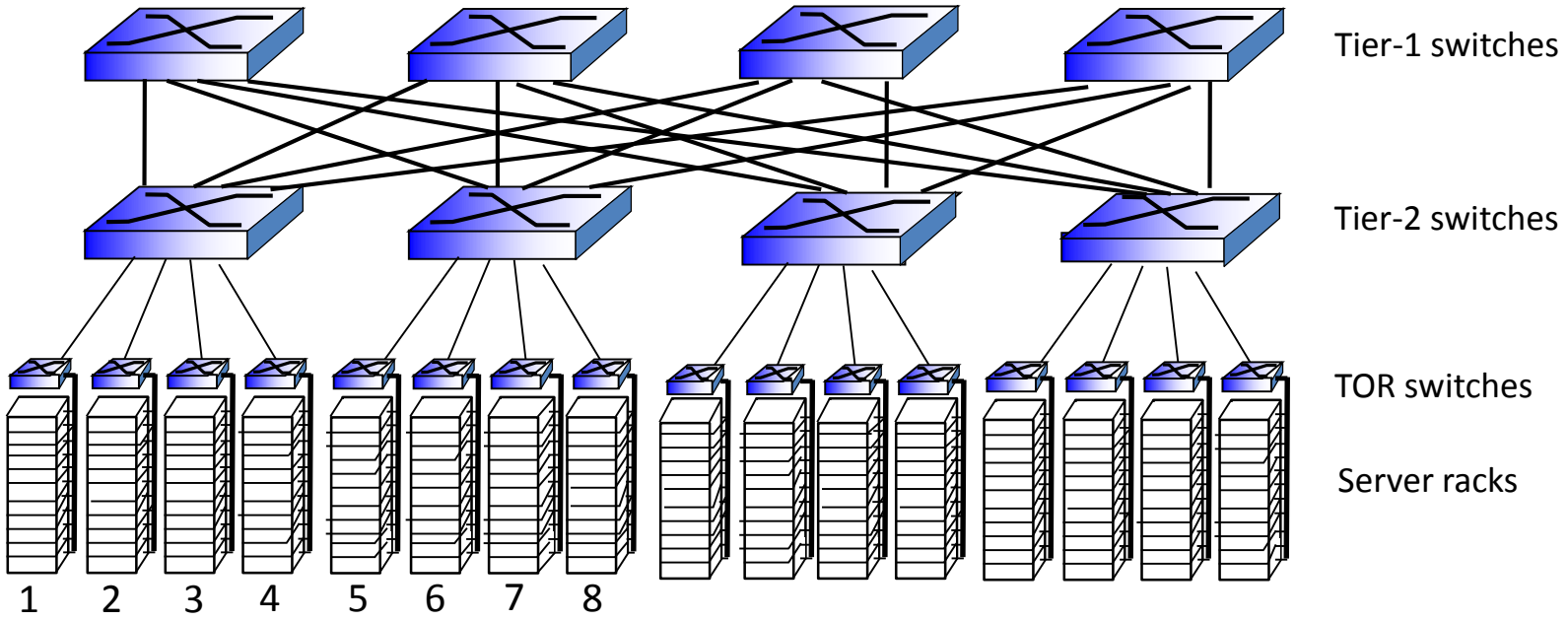
load balancer: routing nell'application-layer

- riceve richieste dai client esterni
- direziona il carico di lavoro nel data center
- restituisce i risultati ai client esterni (nascondendo la struttura dei data center)



Reti di data center

- interconnessione forte tra switch e rack:
  - throughput potenziato tra rack (sono possibili cammini multipli)
  - affidabilità incrementata tramite ridondanza



Livello di link e LAN

5.1 introduzione e servizi

5.2 rilevazione e correzione degli errori

5.3 protocolli di accesso multiplo

5.4 LAN

- indirizzamento, ARP
- Ethernet
- switch
- VLAN

5.5 virtualizzazione di un link: MPLS

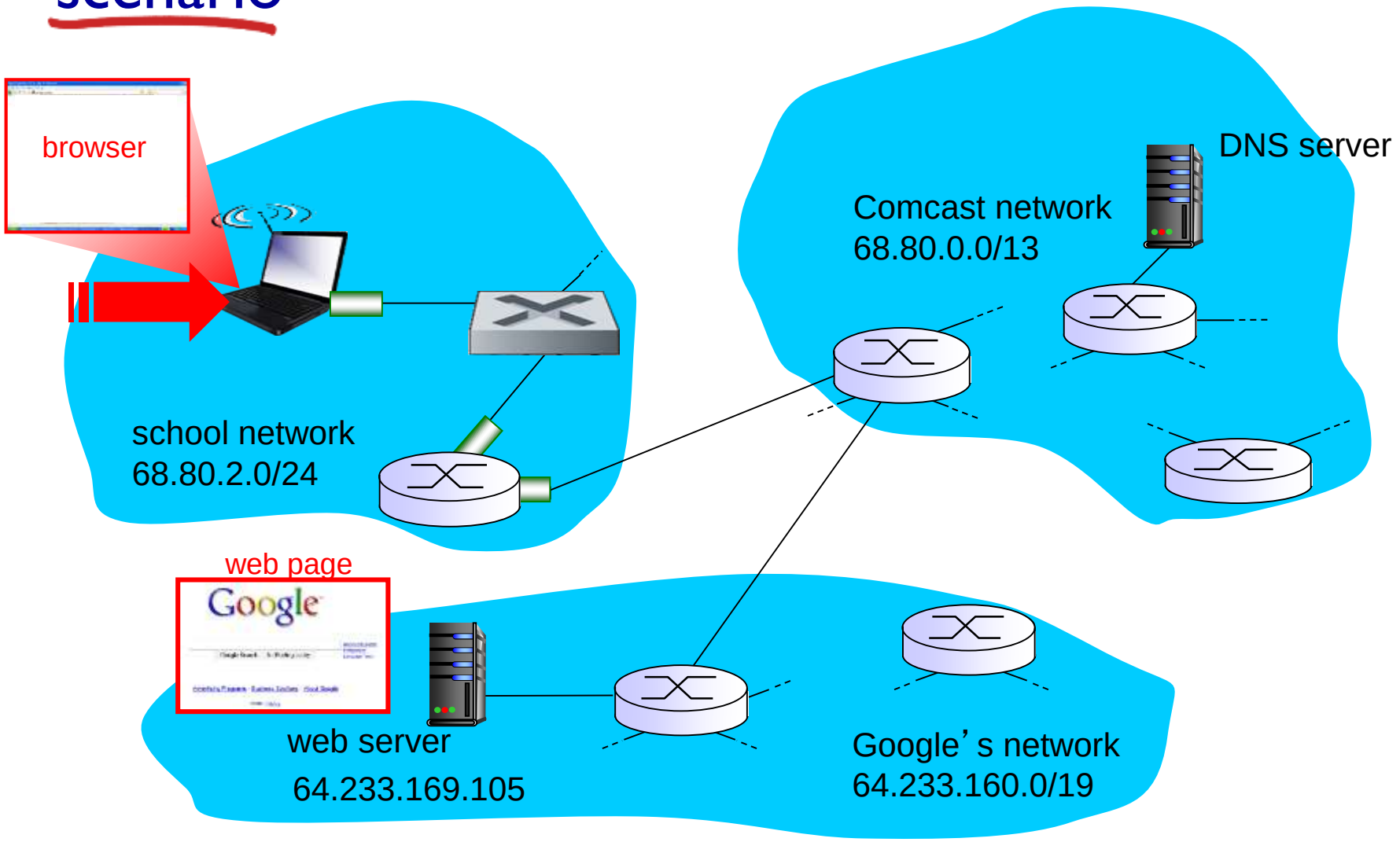
5.6 data center networking

5.7 il cammino di una richiesta web

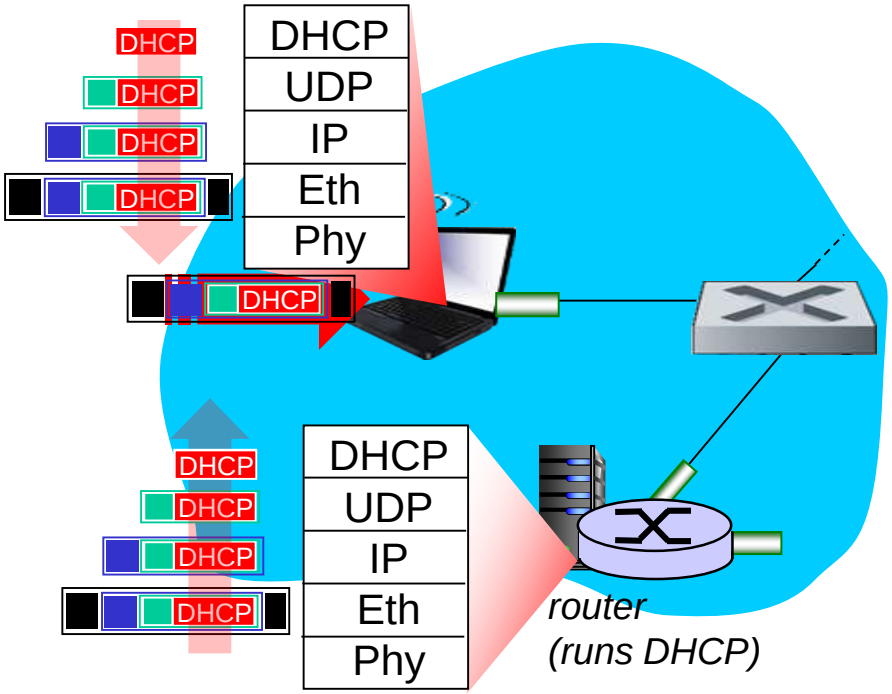
riassumendo: il cammino di una richiesta web

- viaggio lungo lo stack protocollare completo!
  - applicazione, trasporto, rete, link
- mettiamo tutto insieme!
  - *obiettivo:* identificare, rivedere, capire quali protocolli (a tutti i livelli) sono coinvolti in un semplice scenario: la richiesta di una pagina web
  - *scenario:* uno studente collega il suo laptop alle rete del campus, richiede/riceve www.google.com

scenario

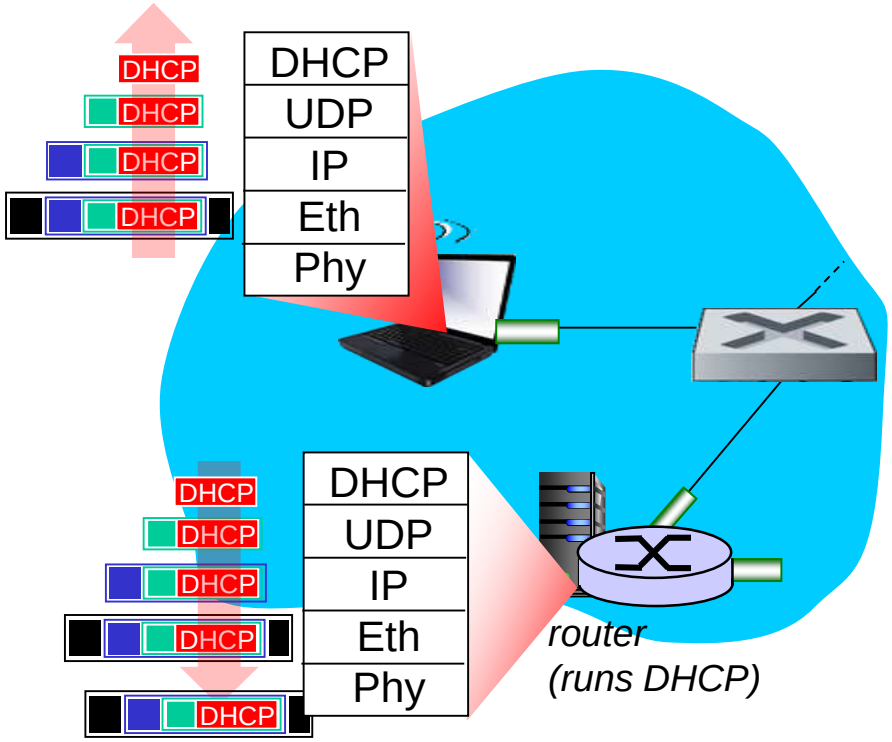

browser
DNS server
Comcast network
68.80.0.0/13
school network
68.80.2.0/24
web page
Google
web server
64.233.169.105
Google's network
64.233.160.0/19

Connessione a Internet



- il laptop ha bisogno del suo indirizzo IP, indirizzo del router di first-hop, indirizzo del server DNS: usa il *DHCP*
- la richiesta DHCP viene *incapsulata* in *UDP*, incapsulata in *IP*, incapsulata in Ethernet *802.3*
- il frame Ethernet in *broadcast* (dest: FFFFFFFFFFFF) sulla LAN, viene ricevuto dal router che fa da server *DHCP*
- il frame Ethernet *demuxed* verso IP, demuxed verso UDP, demuxed verso DHCP

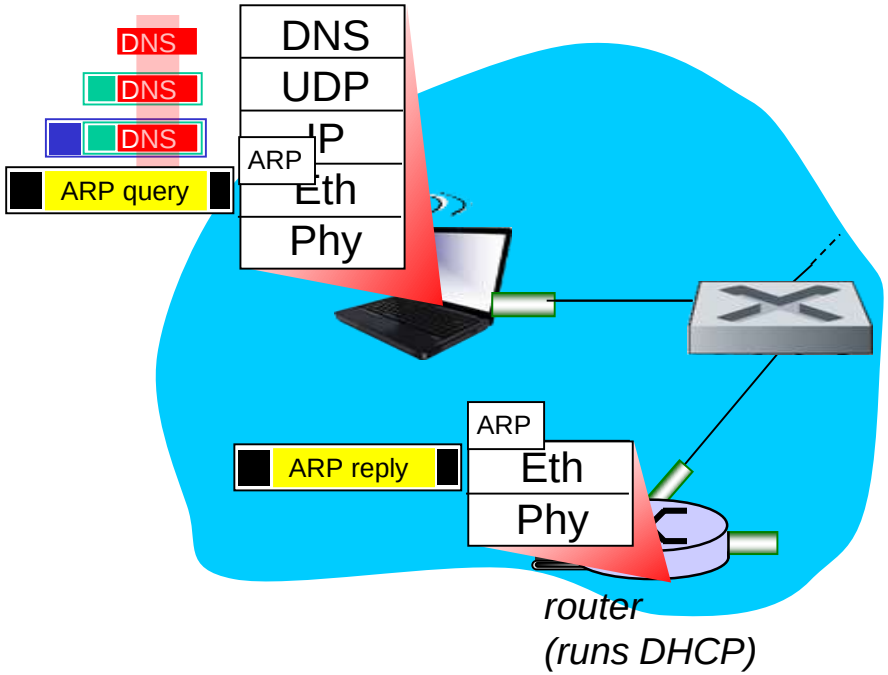
Connessione a Internet



- il server DHCP formula il *DHCP ACK* contenente l'indirizzo IP del client, indirizzo IP del router di first-hop, nome & indirizzo IP del server DNS
- il server DHCP effettua l'incapsulamento, il frame viene inoltrato (*switch learning*) attraverso la LAN, e il client fa demultiplexing
- il client DHCP riceve il reply DHCP ACK

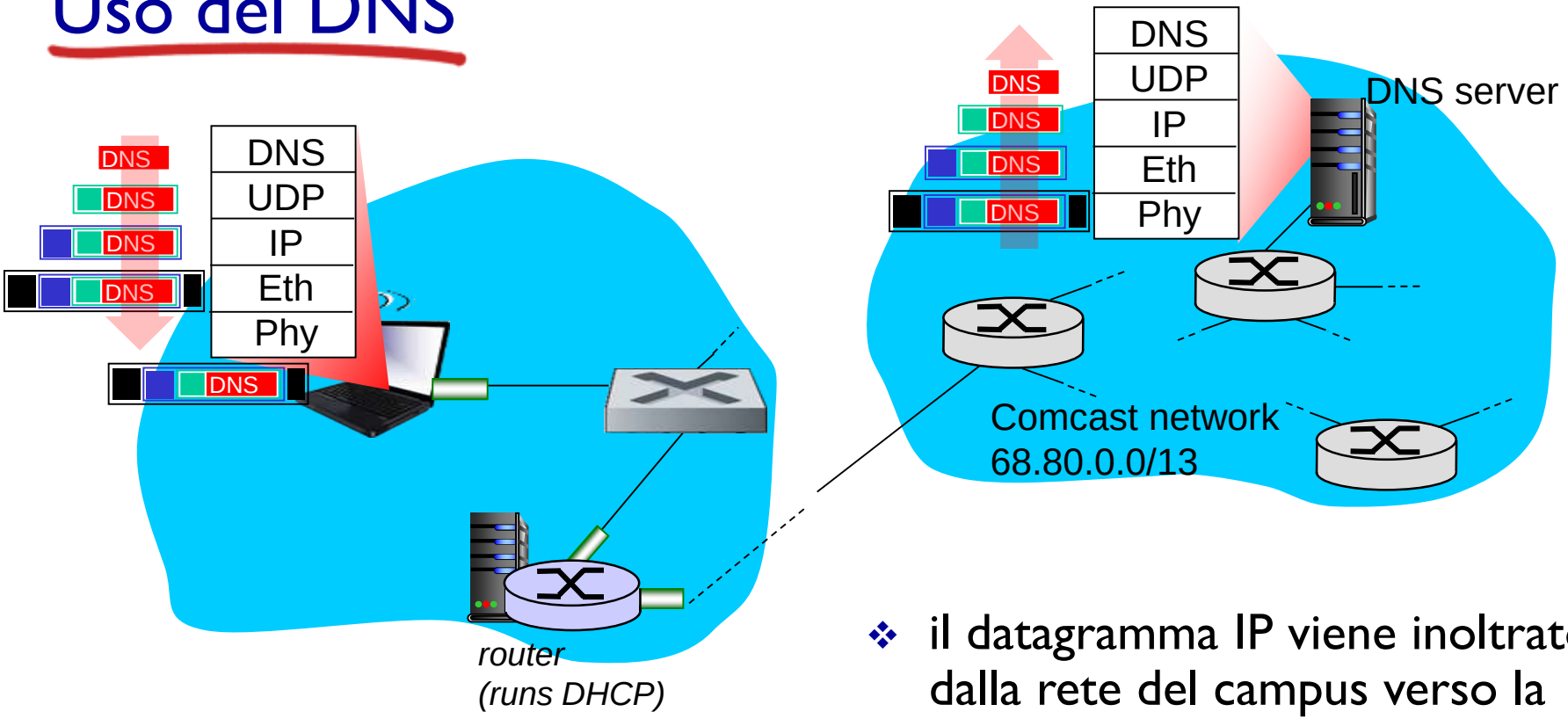
Il client ora ha un indirizzo IP, sa nome & indirizzo IP del server DNS, indirizzo IP del router di first-hop

ARP (prima del DNS, prima di HTTP)



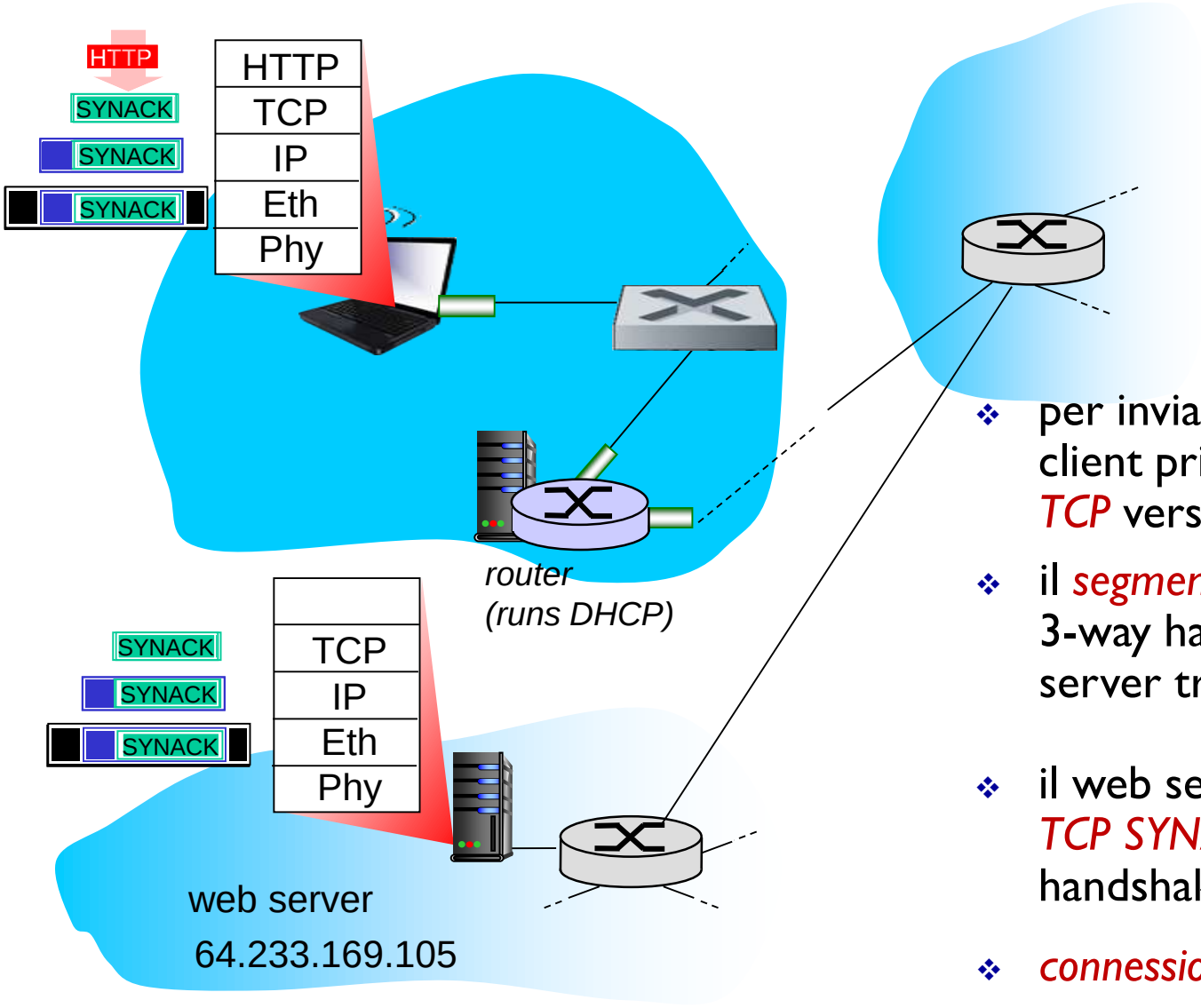
- prima di inviare la richiesta *HTTP*, serve l'indirizzo IP di www.google.com: *DNS*
- viene create la query DNS, incapsulata in UDP, incapsulata in IP, incapsulata in Eth. Per inviare il frame al router, serve l'indirizzo MAC dell'interfaccia del router: *ARP*
- la *query ARP* in broadcast, viene ricevuta dal router, che risponde con un *ARP reply* che dà l'indirizzo MAC dell'interfaccia del router
- il client ora sa l'indirizzo MAC del router di first hop, così può inviare il frame con la query DNS

Uso del DNS



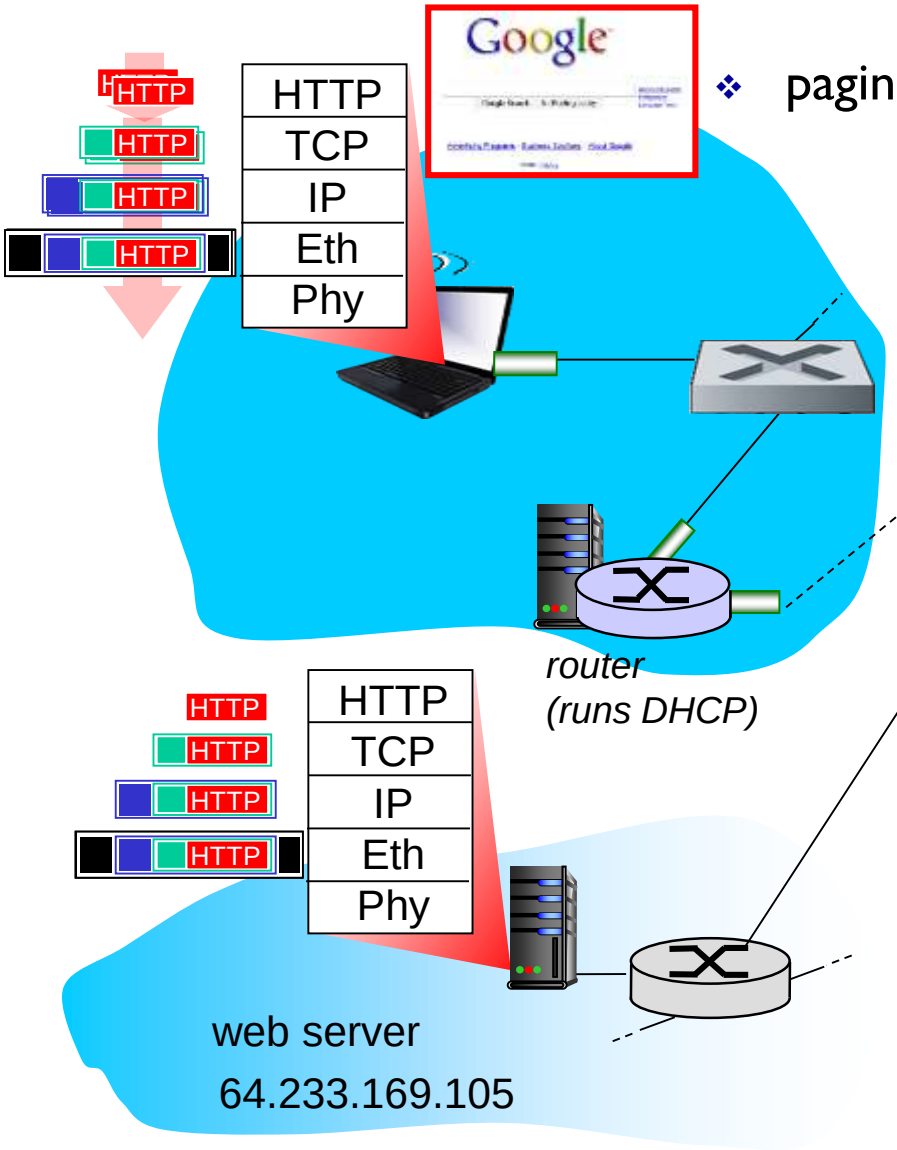
- il datagramma IP contenente la query DNS viene inoltrato tramite lo switch LAN dal client al router di 1st hop
- il datagramma IP viene inoltrato dalla rete del campus verso la rete comcast, tramite routing verso il DNS server (tabelle create da *RIP, OSPF, IS-IS* e/o *BGP*)
- viene 'demuxato' dal server DNS
- il server DNS risponde al client con l'indirizzo IP di www.google.com

Connessione TCP che trasporta l'HTTP



- per inviare l'HTTP request, il client prima apre una *socket TCP* verso il web server
- il *segmento SYN* TCP (step 1 del 3-way handshake) arriva al web server tramite *routing inter-domain*
- il web server risponde con un *TCP SYNACK* (step 2 del 3-way handshake)
- *connessione* TCP *stabilita!*

HTTP request/reply



- pagina web *finalmente (!!!)* mostrata

- l'*HTTP request* viene inviata nella socket TCP
- il datagramma IP contenente l'HTTP request arriva a www.google.com via routing
- il web server risponde con un'*HTTP reply* (contenente la pagina web)
- il datagramma IP contenente l'HTTP reply arriva al client via routing

Capitolo 5: riassunto

- principi per implementare i servizi di trasmissione dati:
  - rilevazione e correzione dell'errore
  - condivisione di un canale broadcast: accesso multiplo
  - indirizzamento a livello di link
- implementazione di varie tecnologie al livello di link
  - Ethernet
  - switched LAN, VLAN
  - reti virtualizzate per il livello di link: MPLS
- il cammino di una richiesta web

Capitolo 5: facciamo il punto

- discesa lungo la pila protocollare *completata* (tranne che per il fisico)
- comprensione dei principi del networking
- ….. ci potremmo fermare qui…. ma ci sono ancora *tanti* argomenti interessanti!
  - wireless
  - multimedia
  - sicurezza
  - gestone della rete