

# Capitolo 5

## Livello di collegamento e reti locali

### Nota per l'utilizzo:

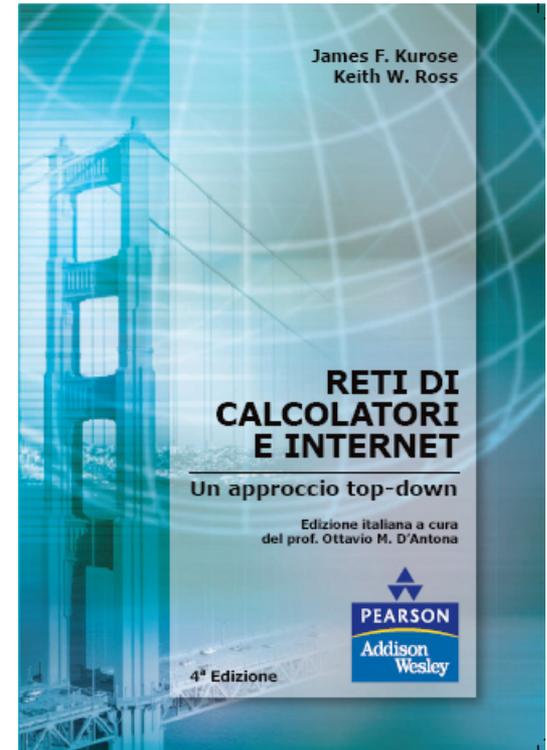
Abbiamo preparato queste slide con l'intenzione di renderle disponibili a tutti (professori, studenti, lettori). Sono in formato PowerPoint in modo che voi possiate aggiungere e cancellare slide (compresa questa) o modificarne il contenuto in base alle vostre esigenze.

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

All material copyright 1996-2007  
J.F Kurose and K.W. Ross, All Rights Reserved



*Reti di calcolatori e Internet:  
Un approccio top-down*

4ª edizione  
Jim Kurose, Keith Ross

Pearson Paravia Bruno Mondadori Spa  
©2008

# Capitolo 5: Livello di collegamento e reti locali

## 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 collegamento
  - Trasferimento affidabile dei dati, controllo del flusso: *già visto!*
- ❑ Istanziamento e implementazione delle varie tecnologie a livello di link.

# Capitolo 5: Livello di collegamento e reti locali

5.1 Livello di collegamento: introduzione e servizi

5.2 Tecniche di rilevazione e correzione degli errori

5.3 Protocolli di accesso multiplo

5.4 Indirizzi a livello di collegamento

5.5 Ethernet

5.6 Switch a livello di collegamento

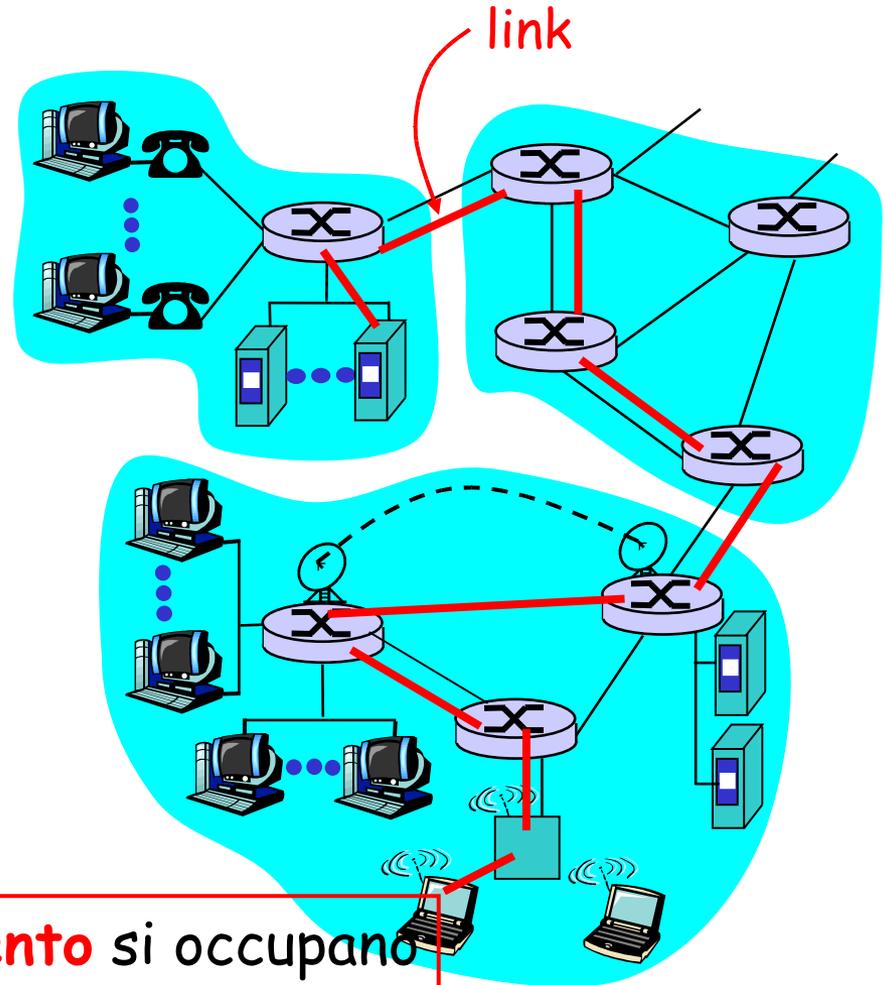
5.7 PPP: protocollo punto-punto

5.8 Canali virtuali: una rete come un livello di link

# Livello di collegamento: introduzione

## Alcuni termini utili:

- host e router sono i **nod**i
- i canali di comunicazione che collegano nodi adiacenti lungo un cammino sono i **collegamenti (link)**
  - collegamenti cablati
  - collegamenti wireless
  - LAN
- Le unità di dati scambiate dai protocolli a livello di link sono chiamate **frame**.



**I protocolli a livello di collegamento** si occupano del trasporto di datagrammi lungo un singolo canale di comunicazione.

# Livello di collegamento

- Un datagramma può essere gestito da diversi protocolli,
- su collegamenti differenti:
  - Es., un datagramma può essere gestito da Ethernet sul primo collegamento, da PPP sull'ultimo e da un protocollo WAN nel collegamento intermedio.
- Anche i servizi erogati dai protocolli del livello di link possono essere differenti:
  - Ad esempio, non tutti i protocolli forniscono un servizio di consegna affidabile.

## Analogia con un tour operator:

- Un viaggio da Princeton a Losanna:
  - taxi: da Princeton all'aeroporto JFK
  - aereo: dal JFK a Ginevra
  - treno: da Ginevra a Losanna
- Turista = **datagramma**
- Ciascuna tratta del trasporto = **collegamento**
- Tipologia del trasporto = **protocollo di link**
- Agente di viaggio = **protocollo di routine**

# Servizi offerti dal livello di link

## □ Framing:

- I protocolli incapsulano i datagrammi del livello di rete all'interno di un frame a livello di link.
- Il protocollo MAC controlla l'accesso al mezzo
- Per identificare origine e destinatario vengono utilizzati indirizzi "MAC"
  - ✂ Diversi rispetto agli indirizzi IP!

## □ Consegna affidabile:

- Come avviene, lo abbiamo già imparato nel Capitolo 3!
- È considerata non necessaria nei collegamenti che presentano un basso numero di errori sui bit (fibra ottica, cavo coassiale e doppino intrecciato)
- È spesso utilizzata nei collegamenti soggetti a elevati tassi di errori (es.: collegamenti wireless)

# Servizi offerti dal livello di collegamento

## ❑ **Controllo di flusso:**

- Evita che il nodo trasmittente saturi quello ricevente.

## ❑ **Rilevazione degli errori:**

- Gli errori sono causati dall'attenuazione del segnale e da rumore elettromagnetico.
- Il nodo ricevente individua la presenza di errori
  - è possibile grazie all'inserimento, da parte del nodo trasmittente, di un bit di controllo di errore all'interno del frame.

## ❑ **Correzione degli errori:**

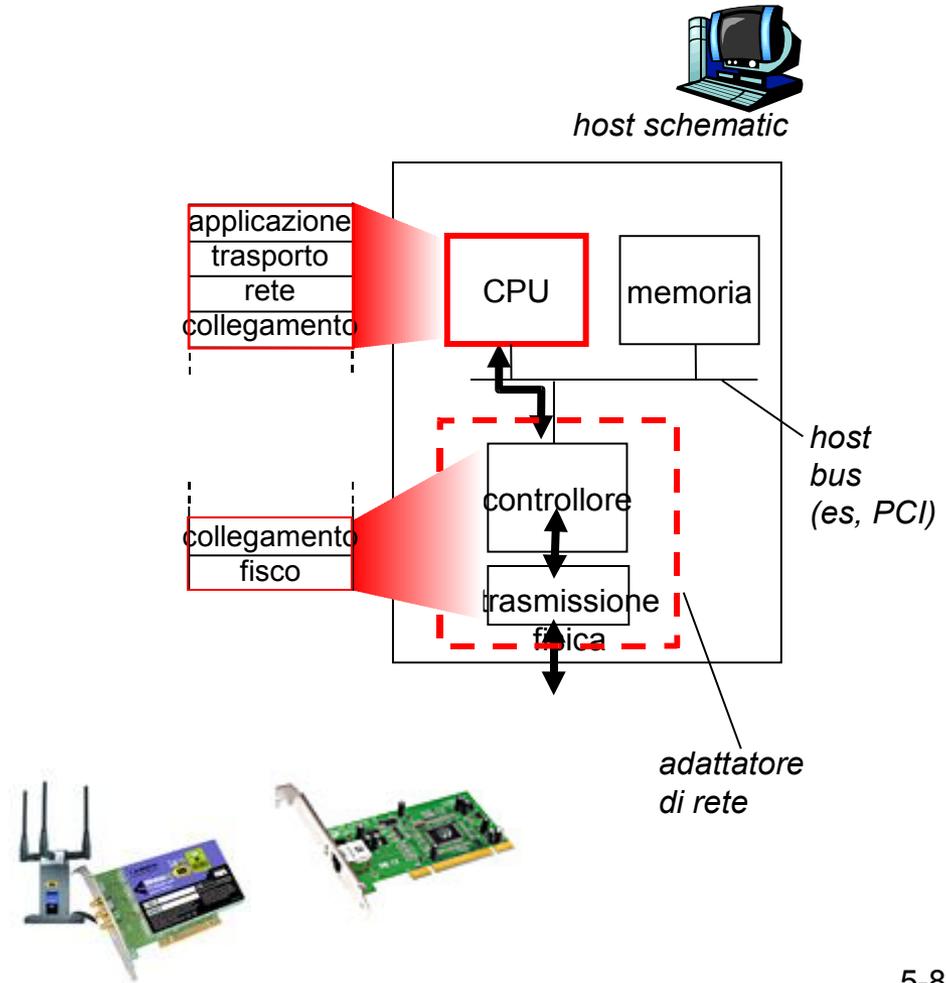
- Il nodo ricevente determina anche il punto in cui si è verificato l'errore, e lo corregge.

## ❑ **Half-duplex e full-duplex**

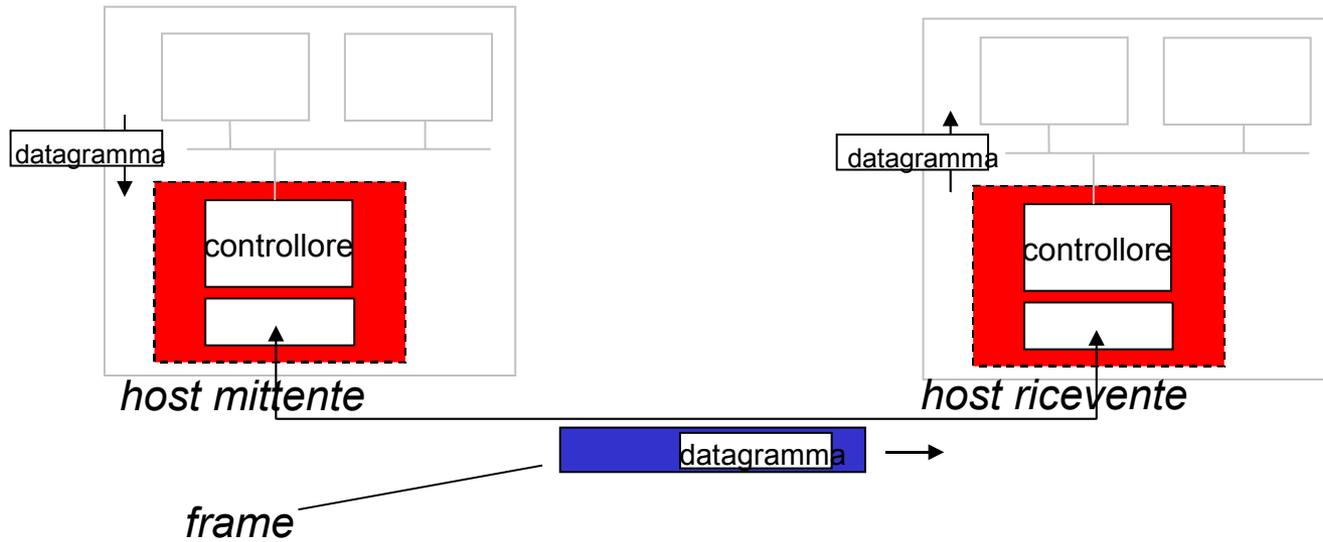
- Nella trasmissione full-duplex gli estremi di un collegamento possono trasmettere contemporaneamente: non in quella half-duplex.

# Dove è implementato il livello di collegamento?

- in tutti gli host
- È realizzato in un adattatore (NIC, *network interface card*)
  - scheda Ethernet, PCMCIA, 802.11
  - Implementa il livello di collegamento e fisico
- è una combinazione di hardware, software e firmware



# Adattatori



## □ Lato mittente:

- Incapsula un datagramma in un frame.
- Imposta il bit rilevazione degli errori, trasferimento dati affidabile, controllo di flusso, etc.

## □ Lato ricevente:

- Individua gli errori, trasferimento dati affidabile, controllo di flusso, etc.
- Estrae i datagrammi e li passa al nodo ricevente

# Capitolo 5: Livello di collegamento e reti locali

5.1 Livello di collegamento: introduzione e servizi

5.2 Tecniche di rilevazione e correzione degli errori

5.3 Protocolli di accesso multiplo

5.4 Indirizzi a livello di collegamento

5.5 Ethernet

5.6 Switch a livello di collegamento

5.7 PPP: protocollo punto-punto

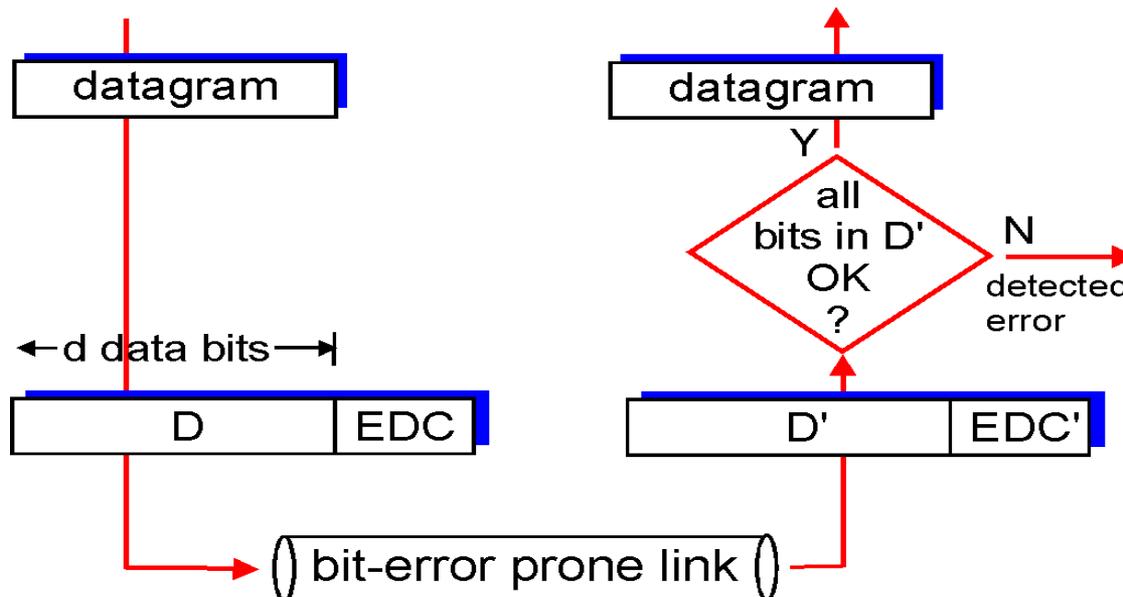
5.8 Canali virtuali: una rete come un livello di link

# Tecniche di rilevazione degli errori

EDC= *Error Detection and Correction*

D = Dati che devono essere protetti da errori e ai quali vengono aggiunti dei bit EDC.

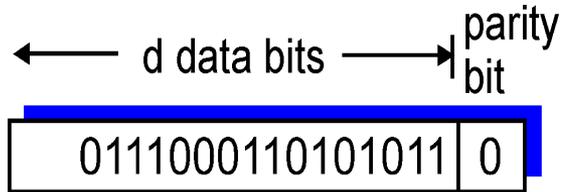
- La rilevazione degli errori non è attendibile al 100%!
  - è possibile che ci siano errori non rilevati
  - per ridurre la probabilità di questo evento, le tecniche più sofisticate prevedono un'elevata ridondanza



# Controllo di parità

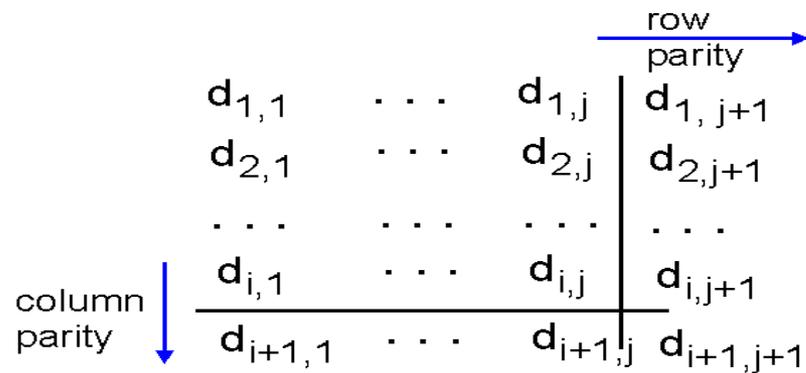
## Unico bit di parità:

Si è verificato almeno un errore in un bit



## Parità bidimensionale:

Individua e *corregge* il bit alterato



101011
111100
011101
0101010

*no errors*

101011
<del>1</del> 1100
011101
0101010

parity error  
 parity error  
*correctable  
 single bit error*

# Checksum di Internet

Obiettivi: rileva gli errori ma viene usata *solo* a livello di trasporto

## Mittente:

- I dati sono trattati come interi da 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:

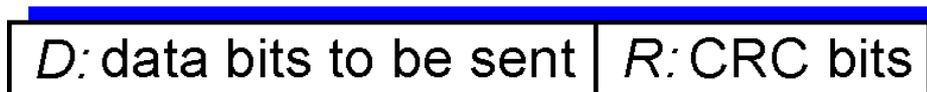
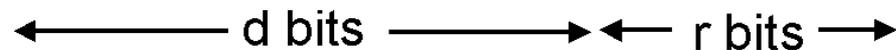
- Il ricevente controlla la checksum.
- Calcola il complemento a 1 della somma dei dati ricevuti e verifica che i bit del risultato siano 1:
  - **NO**, non lo sono: segnala un errore
  - **SÌ** lo sono: non sono stati rilevati errori.

*Ciononostante, ci potrebbero essere altri errori?*

Lo scopriremo in seguito ...

# Controllo a ridondanza ciclica

- ❑ Esamina i dati, **D**, come numeri binari.
- ❑ Origine e destinazione si sono accordati su una stringa di  $r+1$  bit, conosciuta come generatore, **G**.
- ❑ Obiettivi: scegliere  $r$  bit addizionali, **R**, in modo che:
  - $\langle D, R \rangle$  siano esattamente divisibili per  $G$  (modulo 2)
  - Il destinatario conosce  $G$ , e divide  $\langle D, R \rangle$  per  $G$ . Se il resto è diverso da 0 si è verificato un errore!
  - CRC può rilevare errori a raffica inferiori a  $r+1$  bit.
- ❑ Nella pratica è molto usato (ATM, HDCL).



*bit  
pattern*

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

*mathematical  
formula*

# Esempio di CRC

*Vogliamo:*

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

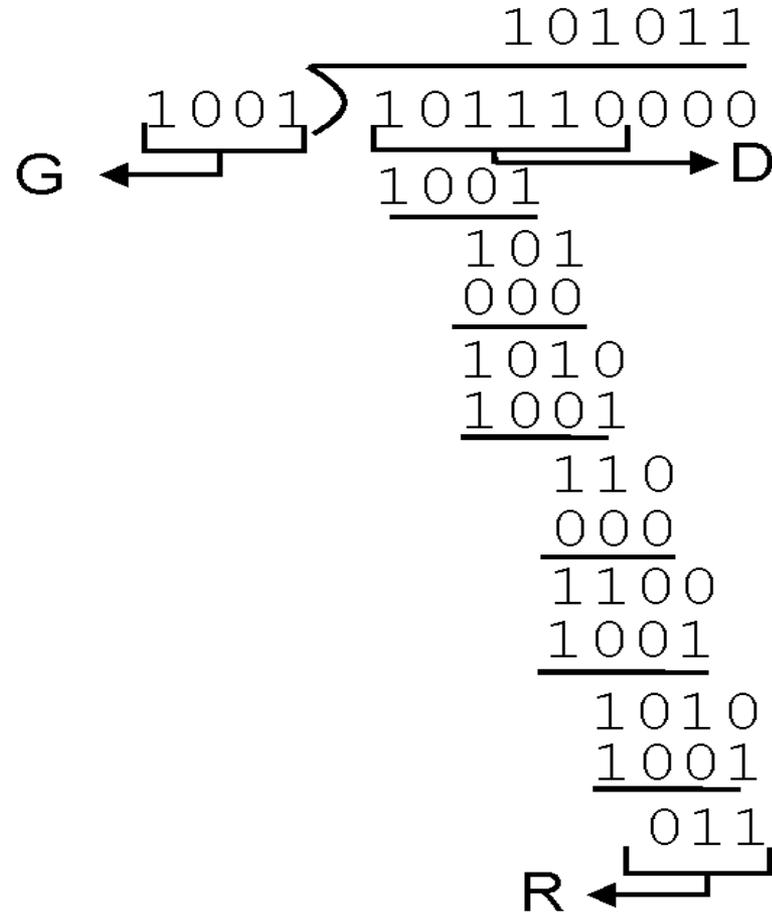
*Ovvero:*

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

*Quindi:*

se dividiamo  $D \cdot 2^r$  per  $G$ , otteniamo il valore  $R$ .

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



# Capitolo 5: Livello di collegamento e reti locali

5.1 Livello di collegamento: introduzione e servizi

5.2 Tecniche di rilevazione e correzione degli errori

5.3 Protocolli di accesso multiplo

5.4 Indirizzi a livello di collegamento

5.5 Ethernet

5.6 Switch a livello di collegamento

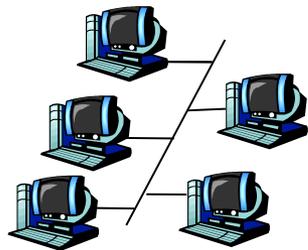
5.7 PPP: protocollo punto-punto

5.8 Canali virtuali: una rete come un livello di link

# Protocolli di accesso multiplo

Esistono due tipi di collegamenti di rete:

- **Collegamento punto-punto (PPP)**
  - Impiegato per connessioni telefoniche.
  - Collegamenti punto-punto tra Ethernet e host.
- **Collegamento broadcast** (cavo o canale condiviso)
  - Ethernet tradizionale
  - HFC in upstream
  - Wireless LAN 802.11



canale cablato  
condiviso



RF condivisa  
(es. 802.11 WiFi)



RF condivisa  
(satellite)



persone a un  
cocktail party  
(rumore, aria condivisi)

# Protocolli di accesso multiplo

- ❑ Connessione a un canale broadcast condiviso.
- ❑ Centinaia o anche migliaia di nodi possono comunicare direttamente su un canale broadcast:
  - Si genera una *collisione* quando i nodi ricevono due o più frame contemporaneamente.

## Protocolli di accesso multiplo

- ❑ Protocolli che fissano le modalità con cui i nodi regolano le loro trasmissioni sul canale condiviso.
- ❑ La comunicazione relativa al canale condiviso deve utilizzare lo stesso canale!
  - non c'è un canale "out-of-band" per la coordinazione

# Protocolli di accesso multiplo ideali

## Canale broadcast con velocità di $R$ bit al sec:

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. Il protocollo è decentralizzato:
  - non ci sono nodi master
  - non c'è sincronizzazione dei clock
4. Il protocollo è semplice.

# Protocolli di accesso multiplo

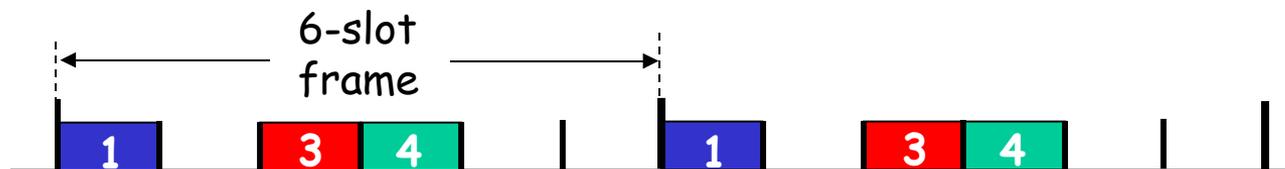
Si possono classificare in una di queste tre categorie:

- ❑ **Protocolli a suddivisione del canale** (*channel partitioning*)
  - Suddivide un canale in "parti più piccole" (slot di tempo, frequenza, codice)
  - Li alloca preses un nodo per utilizzo esclusivo
- ❑ **Protocolli ad accesso casuale** (*random access*)
  - I canali non vengono divisi e si può verificare una collisione.
  - I nodi coinvolti ritrasmettono ripetutamente i pacchetti.
- ❑ **Protocolli a rotazione** ("*taking-turn*")
  - 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: accesso multiplo a divisione di tempo.**

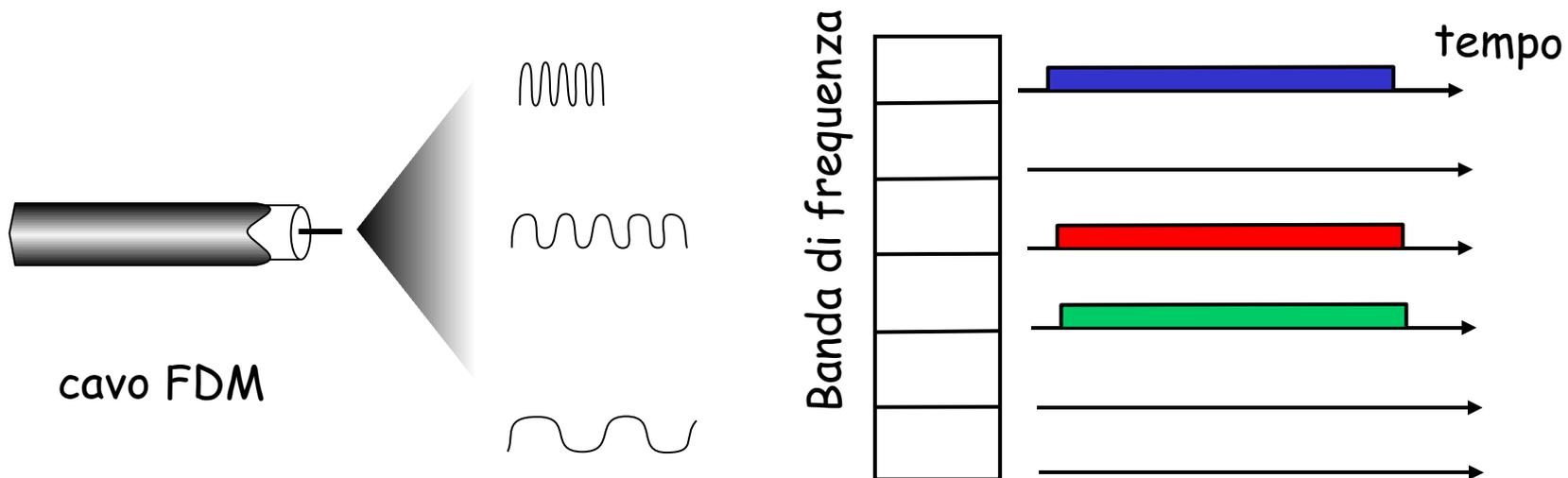
- ❑ Turni per accedere al canale
- ❑ Suddivide il canale condiviso in *intervalli di tempo*.
- ❑ Gli slot non usati rimangono inattivi
- ❑ Esempio: gli slot 1, 3 e 4 hanno un pacchetto, 2, 5 e 6 sono inattivi.



# Protocolli a suddivisione del canale: FDMA

## FDMA: accesso multiplo a divisione di frequenza.

- ❑ Suddivide il canale in bande di frequenza.
- ❑ A ciascuna stazione è assegnata una banda di frequenza prefissata.
- ❑ Esempio: gli slot 1, 3 e 4 hanno un pacchetto, 2, 5 e 6 sono inattivi.



# Protocolli ad accesso casuale

- ❑ Quando un nodo deve inviare un pacchetto:
  - trasmette sempre alla massima velocità consentita dal canale, cioè  $R$  bps
  - non vi è coordinazione a priori tra i nodi
- ❑ Due o più nodi trasmettenti - -> "collisione"
- ❑ **Il protocollo ad accesso casuale** definisce:
  - Come rilevare un'eventuale collisione.
  - Come ritrasmettere se si è verificata una collisione.
- ❑ Esempi di protocolli ad accesso casuale:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

# Slotted ALOHA

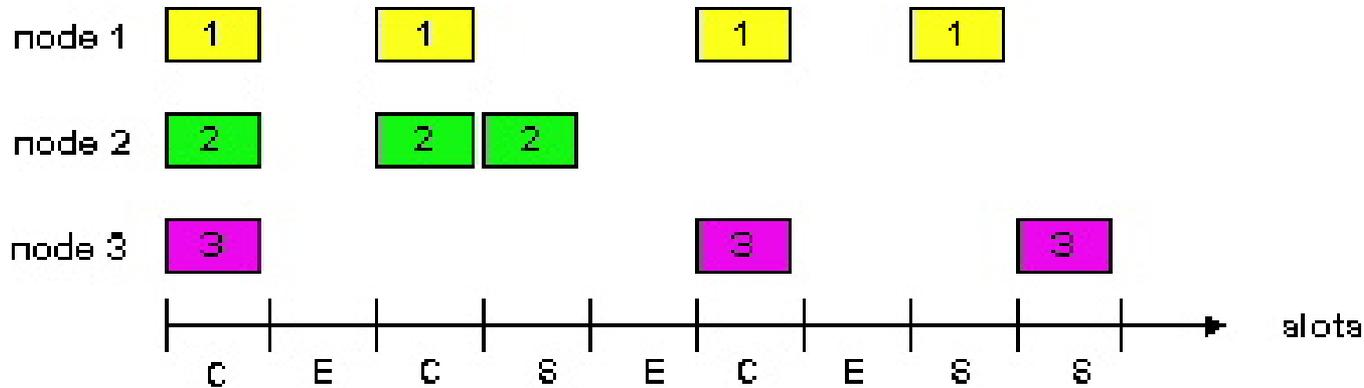
## Assumiamo che:

- Tutti i pacchetti hanno la stessa dimensione.
- Il tempo è 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 in uno slot due o più pacchetti collidono, i nodi coinvolti rilevano l'evento prima del termine dello slot.

## Operazioni:

- Quando a un nodo arriva un nuovo pacchetto da spedire, il nodo attende fino all'inizio dello slot successivo.
  - **Se non si verifica una collisione:** il nodo può trasmettere un nuovo pacchetto nello slot successivo.
  - **Se si verifica una collisione:** il nodo la rileva prima della fine dello slot e ritrasmette con probabilità  $p$  il suo pacchetto durante gli slot successivi.

# Slotted ALOHA



## Pro

- Consente a un singolo nodo di trasmettere continuamente pacchetti alla massima velocità del canale.
- È fortemente decentralizzato, ciascun nodo rileva le collisioni e decide indipendentemente quando ritrasmettere.
- È estremamente semplice.

## Contro

- Una certa frazione degli slot presenterà collisioni e di conseguenza andrà "sprecata".
- Un'alta frazione degli slot rimane vuota, quindi inattiva.

# L'efficienza di Slotted Aloha

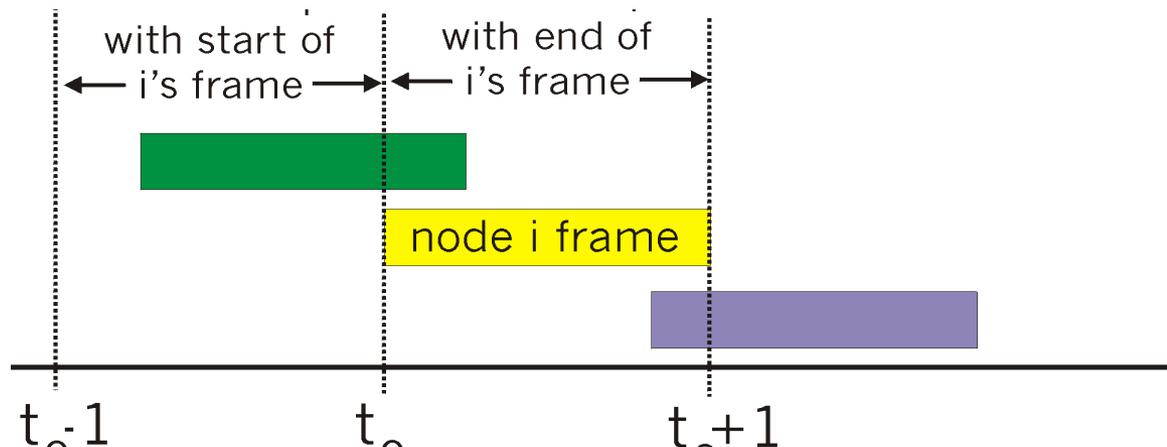
L'**efficienza** è definita come la frazione di slot vincenti in presenza di un elevato numero di nodi attivi, che hanno sempre un elevato numero pacchetti da spedire.

- Supponiamo  $N$  nodi con pacchetti da spedire, ognuno trasmette  $i$  pacchetti in uno slot con probabilità  $p$ .
  - La probabilità di successo di un dato nodo =  $p(1-p)^{N-1}$
  - La probabilità che un nodo arbitrario abbia successo =  $Np(1-p)^{N-1}$
- Per ottenere la massima efficienza con  $N$  nodi attivi, bisogna trovare il valore  $p^*$  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 otterremo  $1/e = 0,37$

*Nel caso migliore:*  
solo il 37% degli slot  
compie lavoro utile.

# ALOHA puro

- ❑ Aloha puro: più semplice, non sincronizzato.
- ❑ Quando arriva il primo pacchetto:
  - lo trasmette immediatamente e integralmente nel canale broadcast.
- ❑ Elevate probabilità di collisione:
  - Il pacchetto trasmesso a  $t_0$  si sovrappone con la trasmissione dell'altro pacchetto inviato in  $[t_0-1, t_0+1]$ .



# L'efficienza di Aloha puro

$P(\text{trasmissione con successo da un dato nodo}) = P(\text{il nodo trasmette}) \cdot$

$P(\text{nessun altro nodo trasmette in } [p_0-1, p_0]) \cdot$

$P(\text{nessun altro nodo trasmette in } [p_0-1, p_0])$

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

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

... scegliendo  $p$  migliore e lasciando  $n \rightarrow$  infinito ...

**Peggior caso**  $S = 0,18$

# Accesso multiplo a rilevazione della portante (CSMA)

**CSMA**: si pone in ascolto prima di trasmettere:

- ❑ Se rileva che il canale è libero, trasmette l'intero pacchetto.
- ❑ Se il canale sta già trasmettendo, il nodo aspetta un altro intervallo di tempo.
  
- ❑ Analogia: se qualcun altro sta parlando, aspettate finché abbia concluso!

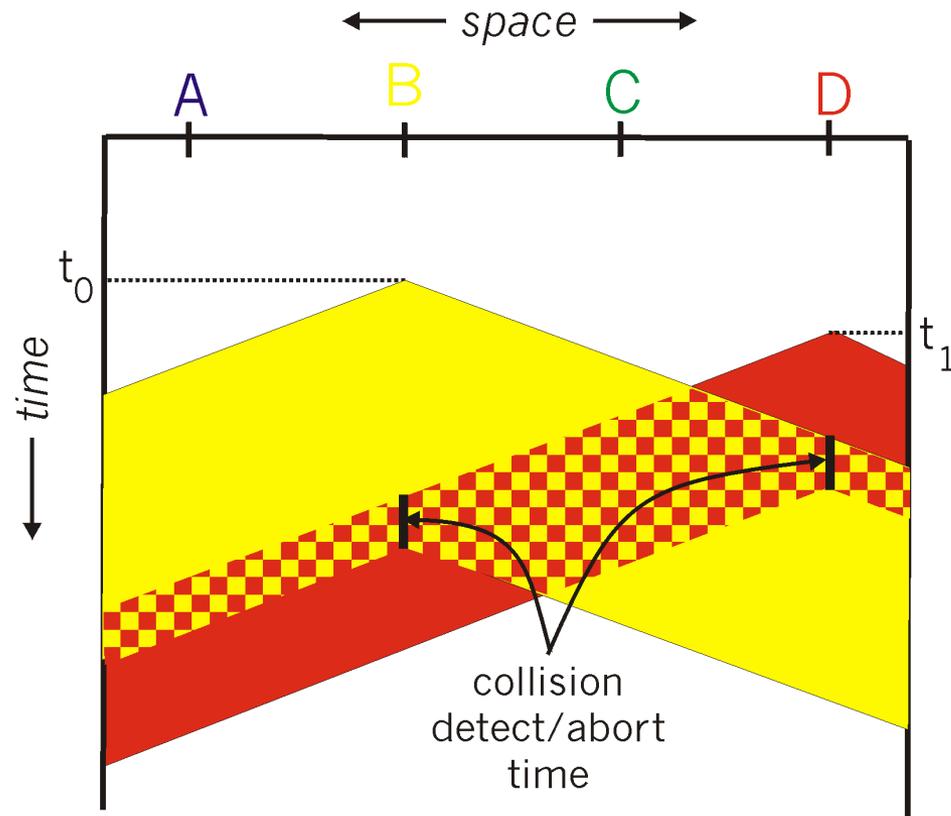


# CSMA/CD (rilevazione di collisione)

**CSMA/CD:** rilevamento della portante differito, come in CSMA:

- Rileva la collisione in poco tempo.
  - Annulla la trasmissione non appena si accorge che c'è un'altra trasmissione in corso.
- Rilevazione della collisione:
- facile nelle LAN cablate.
  - difficile nelle LAN wireless.
- Analogia: un interlocutore educato.

# CSMA/CD (rilevazione di collisione)



# Protocolli MAC a rotazione

## Protocolli MAC a suddivisione del canale:

- Condividono il canale equamente ed efficientemente con carichi elevati.
- Inefficienti con carichi non elevati.

## Protocolli MAC ad accesso casuale:

- Efficienti anche con carichi non elevati: un singolo nodo può utilizzare interamente il canale.
- Carichi elevati: eccesso di collisioni.

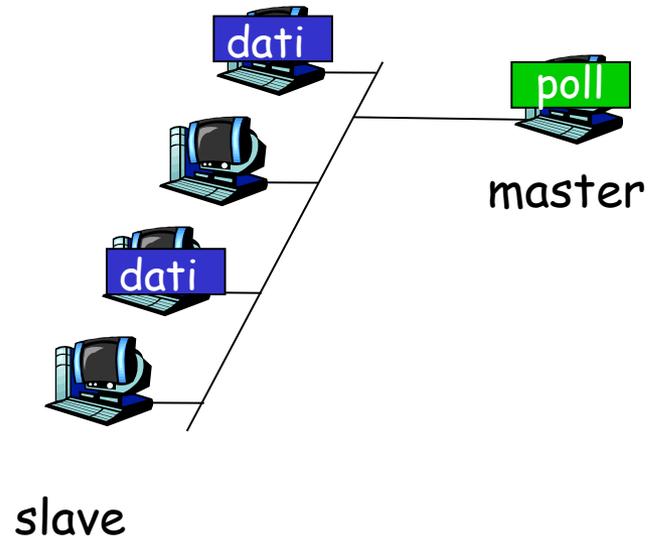
## Protocolli a rotazione

- Prendono il meglio dei due protocolli precedenti!

# Protocolli a rotazione

## Protocollo polling:

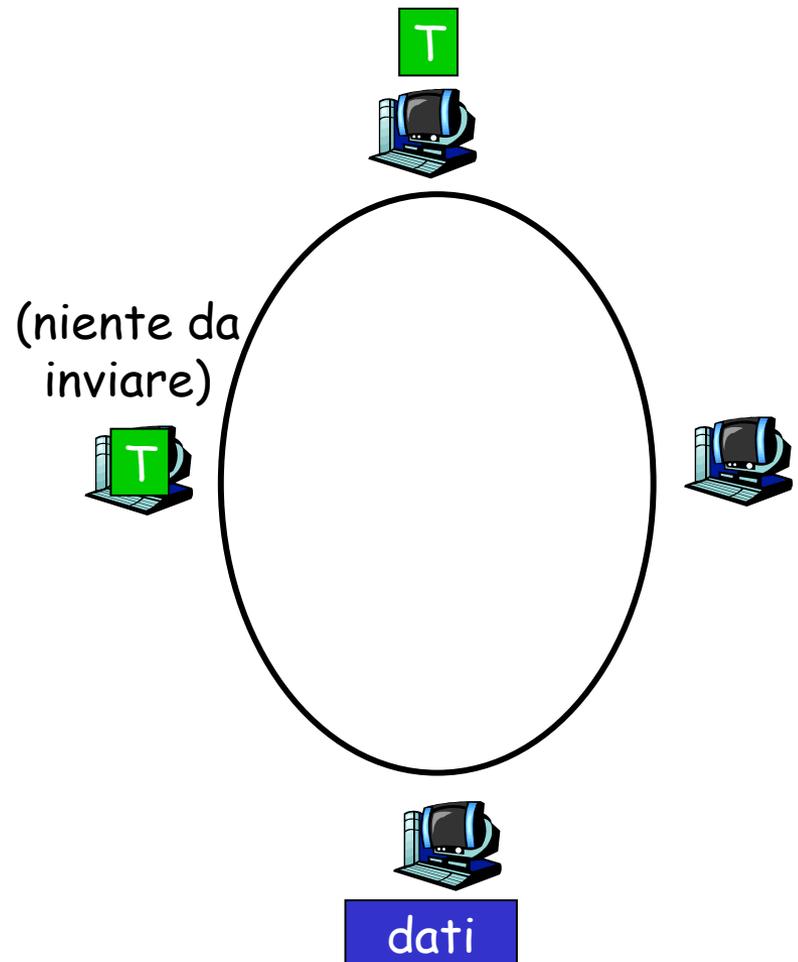
- Un nodo principale sonda "a turno" gli altri.
- In particolare:
  - elimina le collisioni
  - elimina gli slot vuoti
  - ritardo di polling
  - se il nodo principale (master) si guasta, l'intero canale resta inattivo.



# Protocolli a rotazione

## Protocollo token-passing:

- Un messaggio di controllo circola fra i nodi seguendo un ordine prefissato.
- Messaggio di controllo (*token*).
- In particolare:
  - decentralizzato
  - altamente efficiente
  - il guasto di un nodo può mettere fuori uso l'intero canale



# Protocolli: riepilogo

Cosa si può fare con un canale condiviso?

- **Suddivisione del canale** per: tempo, frequenza, codice.
  - TDM, FDM.
- **Suddivisione casuale** (dinamica).
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - Rilevamento della portante: facile in alcune tecnologie (cablate), difficile in altre (wireless)
  - CSMA/CD usato in Ethernet
  - CSMA/CA usato in 802.11
- **A rotazione**
  - Polling con un nodo principale; a passaggio di testimone.
  - Bluetooth, FDDI, IBM Token Ring

# Capitolo 5: Livello di collegamento e reti locali

5.1 Livello di collegamento: introduzione e servizi

5.2 Tecniche di rilevazione e correzione degli errori

5.3 Protocolli di accesso multiplo

5.4 Indirizzi a livello di collegamento

5.5 Ethernet

5.6 Switch a livello di collegamento

5.7 PPP: protocollo punto-punto

5.8 Canali virtuali: una rete come un livello di link

# Indirizzi MAC e ARP

## □ Indirizzo IP a 32 bit:

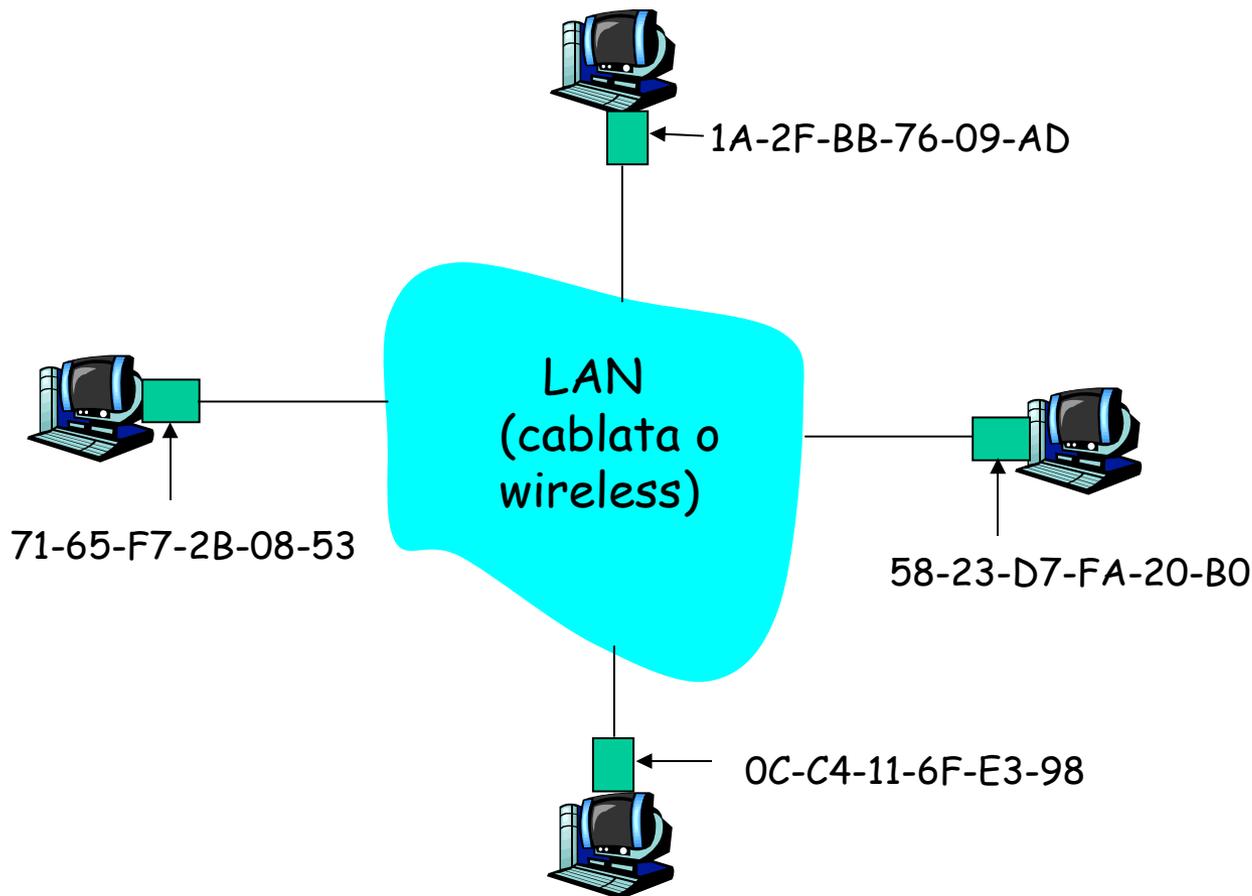
- Indirizzo a *livello di rete*.
- Analogo all'indirizzo postale di una persona: hanno una struttura gerarchica e devono esser aggiornati quando una persona cambia residenza.

## □ Indirizzo MAC (o LAN o fisico o Ethernet):

- Analogo al numero di codice fiscale di una persona: ha una struttura orizzontale e non varia a seconda del luogo in cui la persona si trasferisce.
- Indirizzo a 48 bit (per la maggior parte delle LAN) .

# Indirizzi LAN e ARP

Ciascun adattatore di una LAN ha un indirizzo LAN univoco .



Indirizzo broadcast =  
FF-FF-FF-FF-FF-FF

■ = adattatore

# Indirizzi LAN

- ❑ 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:
  - ❑ (a) Indirizzo MAC: come il codice fiscale di una persona.
  - ❑ (b) Indirizzo IP: come l'indirizzo postale di una persona.
- ❑ Indirizzo orizzontale MAC --> portabilità
  - È possibile spostare una scheda LAN da una LAN a un'altra.
- ❑ Gli indirizzi IP hanno una struttura gerarchica e devono essere aggiornati se spostati.
  - dipendono dalla sottorete IP cui il nodo è collegato.

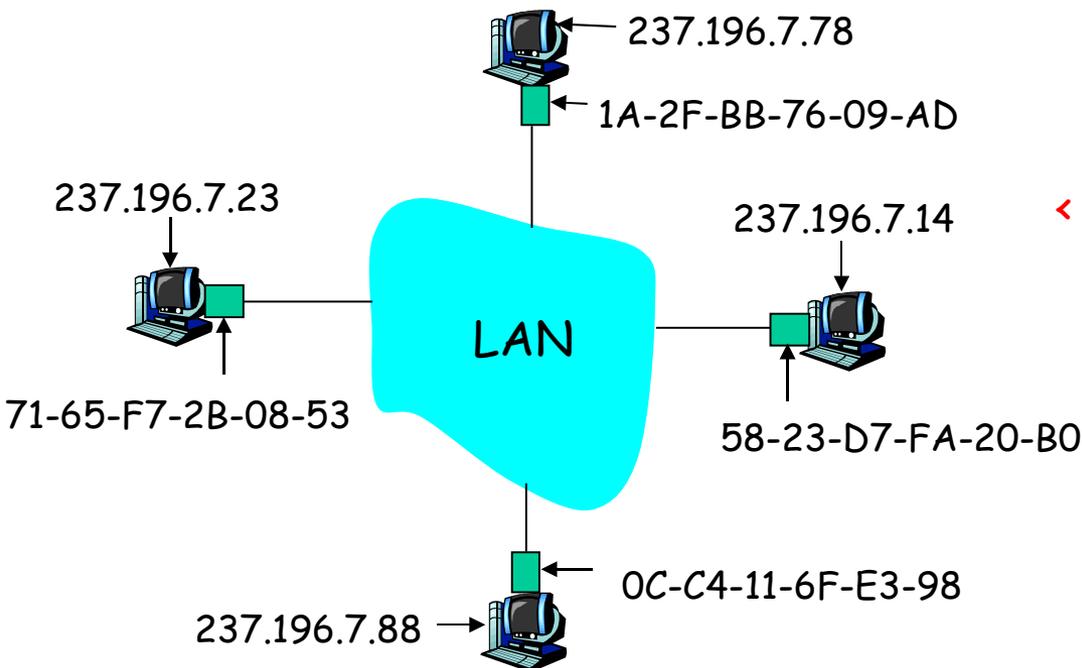
# Protocollo per la risoluzione degli indirizzi (ARP)

Domanda: come si determina l'indirizzo MAC di B se si conosce solo l'indirizzo IP di B?

- Ogni nodo IP (host, router) nella LAN ha una **tabella ARP**.
- Tabella ARP: contiene la corrispondenza tra indirizzi IP e MAC.

< Indirizzo IP; Indirizzo MAC; TTL >

- TTL (tempo di vita): valore che indica quando bisognerà eliminare una data voce nella tabella (il tempo di vita tipico è di 20 min).

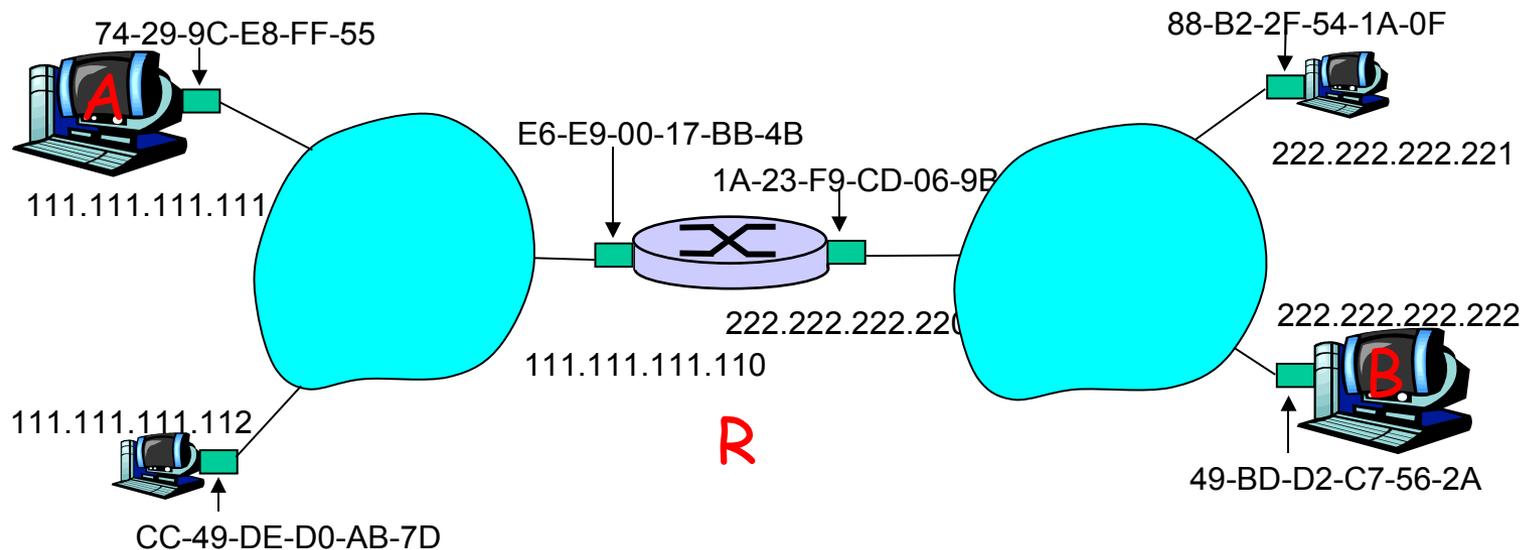


# Protocollo ARP nella stessa sottorete

- *A* vuole inviare un datagramma a *B*, e l'indirizzo MAC di *B* non è nella tabella ARP di *A*.
- *A* trasmette in un pacchetto **broadcast** il messaggio di richiesta ARP, contenente l'indirizzo IP di *B*.
  - Indirizzo MAC del destinatario
  - = FF-FF-FF-FF-FF-FF
  - Tutte le macchine della LAN ricevono una richiesta ARP.
- *B* riceve il pacchetto ARP, e risponde ad *A* comunicandogli il proprio indirizzo MAC.
  - il frame viene inviato all'indirizzo MAC di *A*.
- Il messaggio di richiesta ARP è inviato in un pacchetto broadcast mentre il messaggio di risposta ARP è inviato in un pacchetto standard.
- ARP è "plug-and-play":
  - La tabella ARP di un nodo si costituisce automaticamente e non deve essere configurata dall'amministratore del sistema.

# Invio verso un nodo esterno alla sottorete

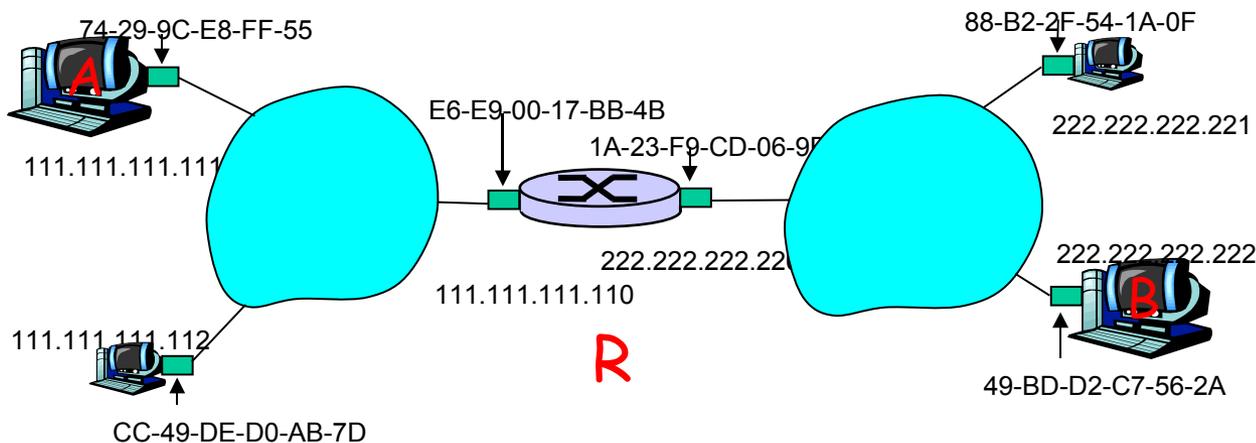
Invio di un datagramma da A a B attraverso R, ipotizzando che A conosca l'indirizzo IP di B.



- Due tabelle ARP nel router R, una per ciascuna rete IP (LAN).

- ❑ A crea un datagramma con origine A, e destinazione B.
- ❑ A usa ARP per ottenere l'indirizzo MAC di R.
- ❑ A crea un collegamento a livello di rete con l'indirizzo MAC di destinazione di R, il frame contiene il datagramma IP da A a B.
- ❑ L'adattatore di A invia il datagramma.
- ❑ L'adattatore di R riceve il datagramma.
- ❑ R rimuove il datagramma IP dal frame Ethernet, e vede che la sua destinazione è B.
- ❑ R usa ARP per ottenere l'indirizzo MAC di B.
- ❑ R crea un frame contenente il datagramma IP da A a B IP e lo invia a B.

Questo esempio è **molto** importante!  
Siete sicuri di averlo compreso bene?



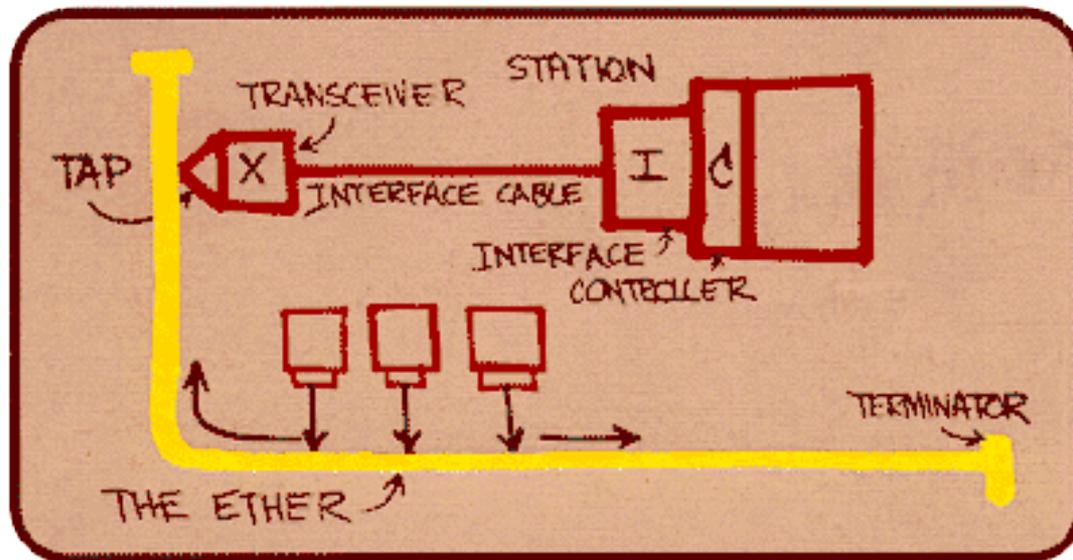
# Capitolo 5: Livello di collegamento e reti locali

- 5.1 Livello di collegamento: introduzione e servizi
- 5.2 Tecniche di rilevazione e correzione degli errori
- 5.3 Protocolli di accesso multiplo
- 5.4 Indirizzi a livello di collegamento
- 5.5 Ethernet
- 5.6 Switch a livello di collegamento
- 5.7 PPP: protocollo punto-punto
- 5.8 Canali virtuali: una rete come un livello di link

# Ethernet

Detiene una posizione dominante nel mercato delle LAN cablate.

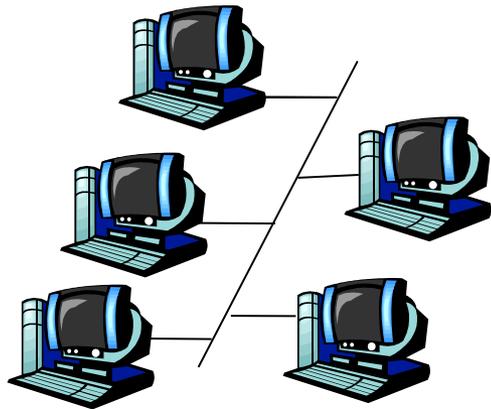
- ❑ È stata la prima LAN ad alta velocità con vasta diffusione.
- ❑ Più semplice e meno costosa di token ring, FDDI e ATM.
- ❑ Sempre al passo dei tempi con il tasso trasmissivo: 10 Mbps - 10 Gbps



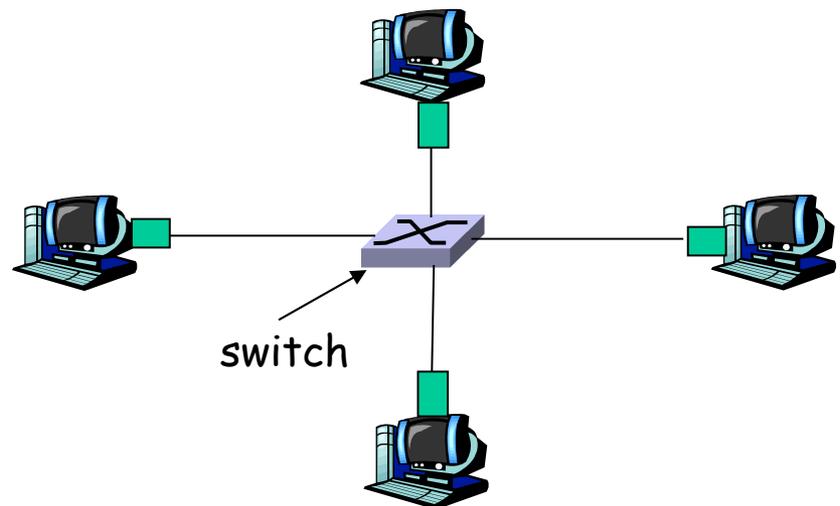
Il progetto originale di Bob Metcalfe che portò allo standard Ethernet.

# Topologia a stella

- ❑ La topologia a bus era diffusa fino alla metà degli anni 90.
- ❑ Quasi tutte le odierne reti Ethernet sono progettate con topologia a stella.
- ❑ Al centro della stella è collocato un hub o commutatore (*switch*).
- ❑ Ciascun nodo esegue un protocollo Ethernet separato e non entra in collisione con gli altri



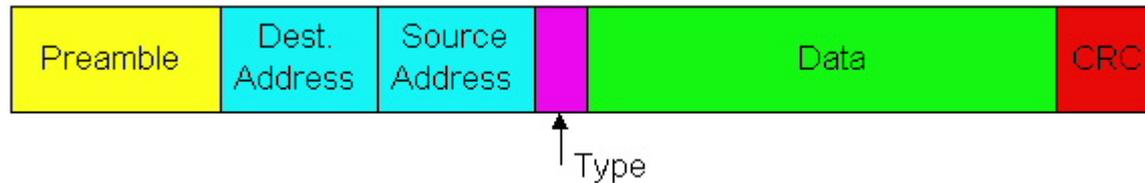
Bus: cavo coassiale



Topologia a stella

# Struttura dei pacchetti Ethernet

L'adattatore trasmittente incapsula i datagrammi IP in un **pacchetto Ethernet**.

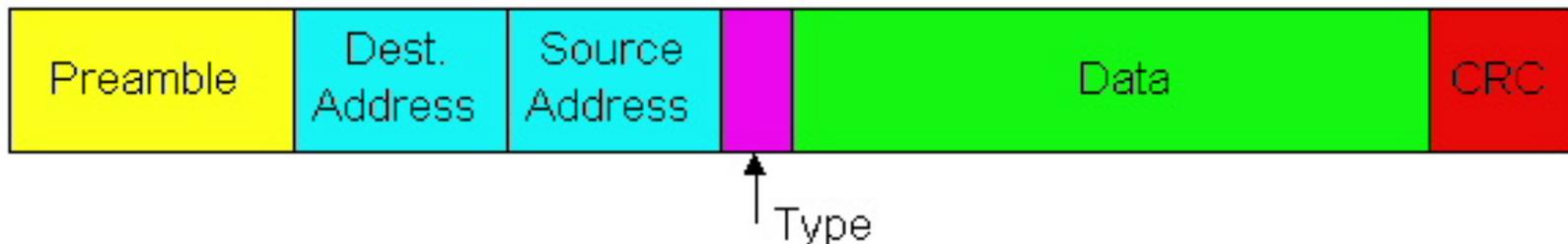


## Preambolo:

- ❑ I pacchetti Ethernet iniziano con un campo di otto byte: sette hanno i bit 10101010 e l'ultimo è 10101011.
- ❑ Servono per "attivare" gli adattatori dei riceventi e sincronizzare i loro orologi con quello del trasmittente.

# Struttura dei pacchetti Ethernet

- ❑ **Indirizzi:** 6 byte
  - Quando un adattatore riceve un pacchetto contenente l'indirizzo di destinazione o con l'indirizzo broadcast (es.: un pacchetto ARP), trasferisce il contenuto del campo dati del pacchetto al livello di rete.
  - I pacchetti con altri indirizzi MAC vengono ignorati.
- ❑ **Campo tipo:** consente a Ethernet di supportare vari protocolli di rete (in gergo questa è la funzione di "multiplexare" i protocolli).
- ❑ **Controllo CRC:** consente all'adattatore ricevente di rilevare la presenza di un errore nei bit del pacchetto.



# Servizio senza connessione non affidabile

- ❑ **Senza connessione:** non è prevista nessuna forma di handshake preventiva con il destinatario prima di inviare un pacchetto.
- ❑ **Non affidabile:** l'adattatore ricevente non invia un riscontro né se un pacchetto supera il controllo CRC né in caso contrario.
  - Il flusso dei datagrammi che attraversano il livello di rete può presentare delle lacune.
  - L'applicazione può rilevare le lacune se viene impiegato TCP.
  - Altrimenti, potrebbe accusare problemi a causa dell'incompletezza dei dati.

# Fasi operative del protocollo CSMA/CD

1. L'adattatore riceve un datagramma di rete dal nodo cui è collegato e prepara un pacchetto Ethernet.
2. Se il canale è inattivo, inizia la trasmissione. Se il canale risulta occupato, resta in attesa fino a quando non rileva più il segnale.
3. Verifica, durante la trasmissione, la presenza di eventuali segnali provenienti da altri adattatori. Se non ne rileva, considera il pacchetto spedito.
4. Se rileva segnali da altri adattatori, interrompe immediatamente la trasmissione del pacchetto e invia un segnale di disturbo (*jam*).
5. L'adattatore rimane in attesa. Quando riscontra l' $n$ -esima collisione consecutiva, stabilisce un valore  $k$  tra  $\{0, 1, 2, \dots, 2^m - 1\}$ . L'adattatore aspetta un tempo pari a  $K$  volte 512 bit e ritorna al Passo 2.

# Protocollo CSMA/CD di Ethernet

**Segnale di disturbo (*jam*):** la finalità è di avvisare della collisione tutti gli altri adattatori che sono in fase trasmissiva; 48 bit.

**Bit di tempo:** corrisponde a 0,1 microsec per Ethernet a 10 Mbps; per  $K=1023$ , il tempo di attesa è di circa 50 msec.

Guardate e interagite  
con l'applet Java  
sul sito web:  
Altamente raccomandato !

**Attesa esponenziale:**

- **Obiettivo:** l'adattatore prova a stimare quanti sono gli adattatori coinvolti.
  - Se sono numerosi il tempo di attesa potrebbe essere lungo.
- **Prima collisione:** sceglie  $K$  tra  $\{0,1\}$ ; il tempo di attesa è pari a  $K$  volte 512 bit.
- **Dopo la seconda collisione:** sceglie  $K$  tra  $\{0,1,2,3\}$ ...
- **Dopo dieci collisioni,** sceglie  $K$  tra  $\{0,1,2,3,4,\dots,1023\}$ .

# Efficienza di Ethernet

- $T_{prop}$  = tempo massimo che occorre al segnale per propagarsi fra una coppia di adattatori.
- $t_{trasm}$  = tempo necessario per trasmettere un pacchetto della maggior dimensione possibile.

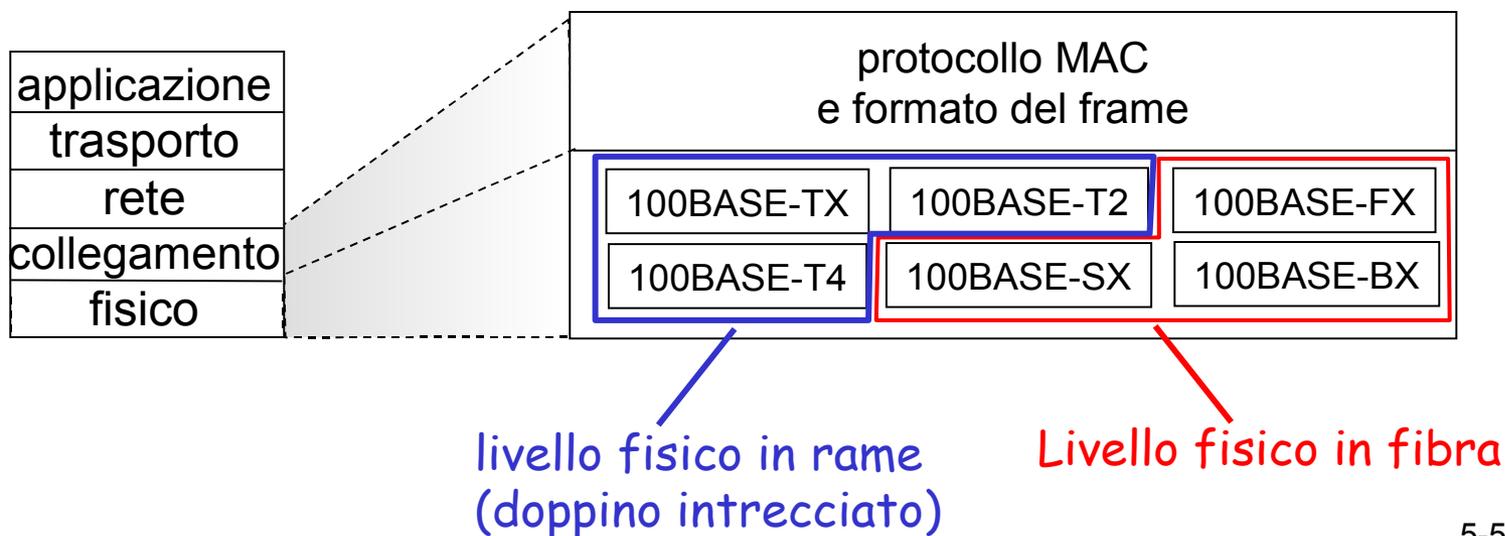
$$\text{efficienza} = \frac{1}{1 + 5t_{prop} / t_{trasm}}$$

- Si evince che quando  $t_{prop}$  tende a 0, l'efficienza tende a 1.
- Al crescere di  $t_{trans}$ , l'efficienza tende a 1.
- Molto meglio di ALOHA: decentralizzato, semplice, e poco costoso.

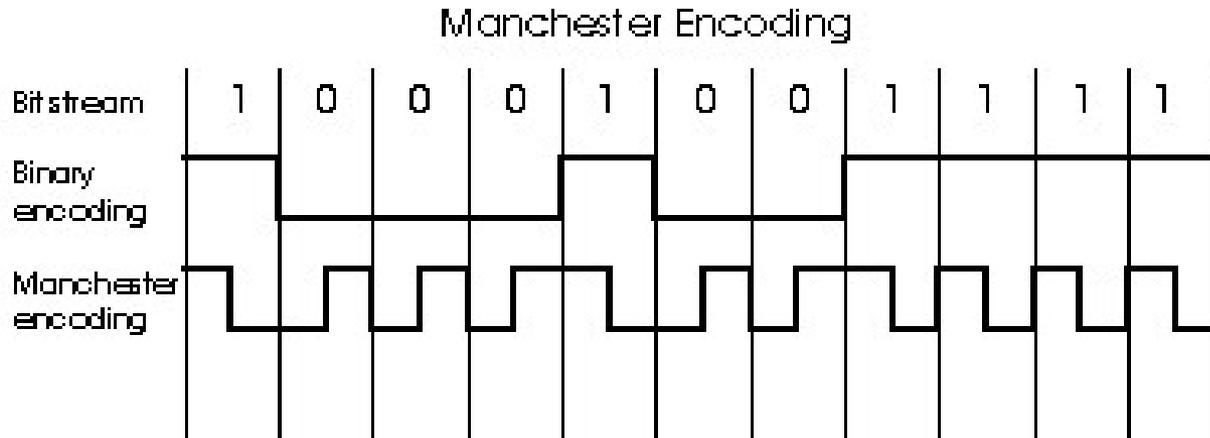
# Ethernet 802.3 : livelli di collegamento e fisico

## □ *multi* differenti standard Ethernet

- protocollo MAC e formato del frame standard
- differenti velocità: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
- differenti mezzi trasmissivi del livello fisico: fibra, cavo



# Codifica Manchester



- ❑ Usata in 10BaseT
- ❑ Durante la ricezione di ciascun bit si verifica una transizione.
- ❑ Permette di sincronizzare gli orologi degli adattatori trasmettenti e riceventi.
  - Non c'è bisogno di un clock centralizzato e globale tra i nodi!
- ❑ La codifica Manchester è un'operazione del livello fisico!

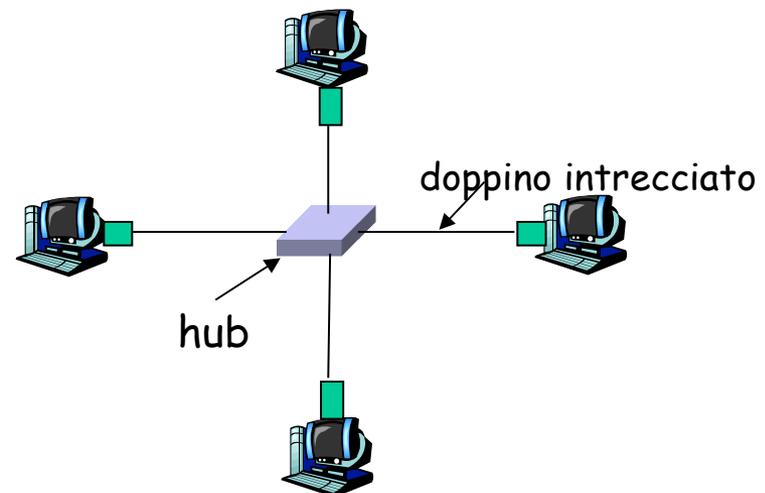
# Capitolo 5: Livello di collegamento e reti locali

- 5.1 Livello di collegamento: introduzione e servizi
- 5.2 Tecniche di rilevazione e correzione degli errori
- 5.3 Protocolli di accesso multiplo
- 5.4 Indirizzi a livello di collegamento
- 5.5 Ethernet
- 5.6 Switch a livello di collegamento
- 5.7 PPP: protocollo punto-punto
- 5.8 Canali virtuali: una rete come un livello di link

# Hub

L'hub (ripetitore) è un dispositivo "stupido" che opera sui singoli bit:

- all'arrivo di un bit, l'hub lo riproduce incrementandone l'energia e lo trasmette attraverso tutte le sue altre interfacce.
- non implementa la rilevazione della portante né CSMA/CD
- ripete il bit entrante su tutte le interfacce uscenti anche se su qualcuna di queste c'è un segnale
- trasmette in broadcast, e quindi ciascun adattatore può sondare il canale per verificare se è libero e rilevare una collisione mentre trasmette
- fornisce aspetti di gestione di rete.

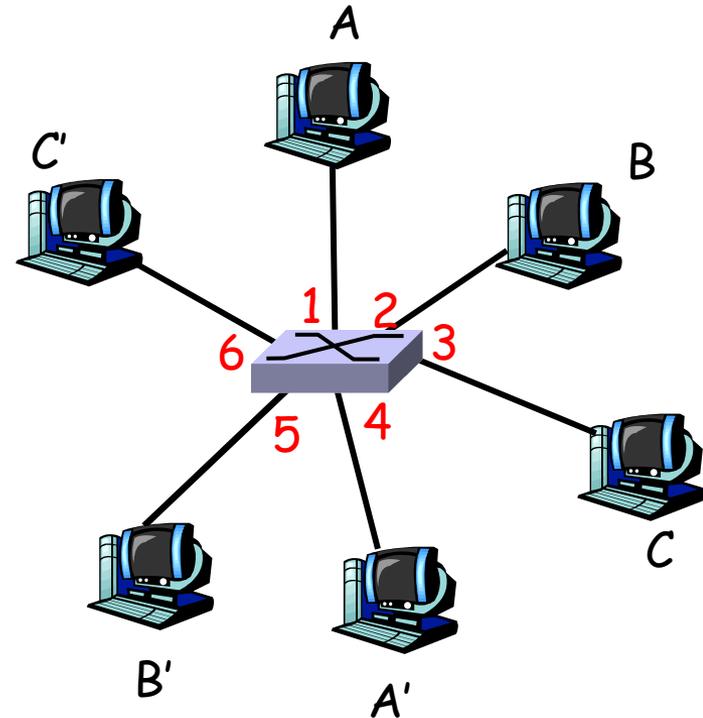


# Switch

- **Dispositivo del livello di link: più intelligente di un hub, svolge un ruolo *attivo***
  - Filtra e inoltra i pacchetti Ethernet.
  - Esamina l'indirizzo di destinazione e lo invia all'interfaccia corrispondente alla sua destinazione.
  - Quando un pacchetto è stato inoltrato nel segmento, usa CSMA/CD per accedere al segmento.
- **Trasparente**
  - Gli host sono inconsapevoli della presenza di switch.
- **Plug-and-play, autoapprendimento**
  - Gli switch non hanno bisogno di essere configurati.

# Switch: consente più trasmissioni simultanee

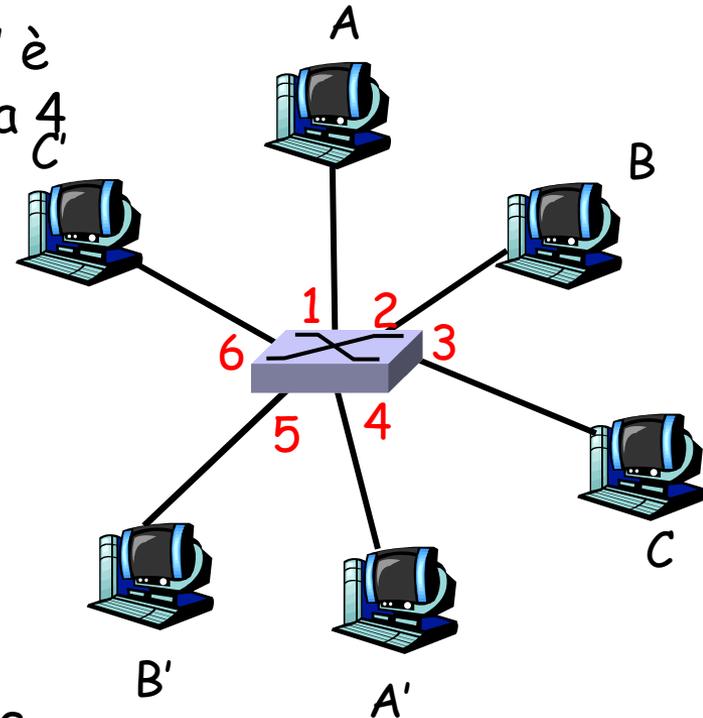
- Gli host hanno collegamenti dedicati e diretti con lo switch
- Gli switch bufferizzano i pacchetti
- Il protocollo Ethernet è usato su ciascun collegamento in entrata, ma non si verificano collisioni; full duplex
- **switching**: da A ad A' e da B a B' simultaneamente, senza collisioni
  - Non possibile con gli hub "stupidi"



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

# Tabella di commutazione

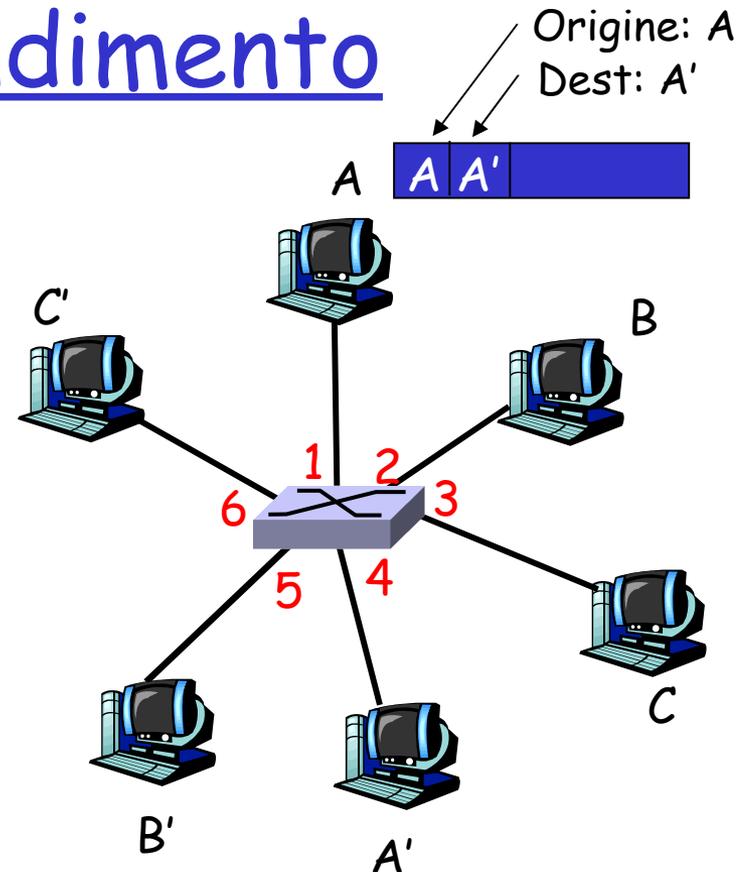
- **D:** come fa lo switch a sapere che A' è raggiungibile attraverso l'interfaccia 4, e B' attraverso l'interfaccia 5?
- **R:** ogni switch ha una **tabella di commutazione** (*switch table*), e ciascuna voce:
  - (indirizzo MAC del nodo, interfaccia che conduce al nodo, time stamp)
- Assomiglia a una tabella d'instradamento!
- **D:** come si creano e si mantengono le voci di una tabella di commutazione?
  - Come in un protocollo di instradamento?



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

# Switch: autoapprendimento

- Lo switch *apprende* quali nodi possono essere raggiunti attraverso determinate interfacce
  - quando riceve un pacchetto, lo switch "impara" l'indirizzo del mittente
  - registra la coppia mittente/indirizzo nella sua tabella di commutazione



Indir. MAC	Interfaccia	TTL
A	1	60

*Tabella di commutazione (inizialmente vuota)*

# Switch: filtraggio e inoltra

## Quando uno switch riceve un pacchetto:

1. Registra il collegamento associato con l'host mittente
2. Indicizza la tabella utilizzando gli indirizzi MAC

**if** entry found for destination  
**then**{

**if** dest on segment from which frame arrived  
**then** drop the frame

**else** forward the frame on interface indicated

}

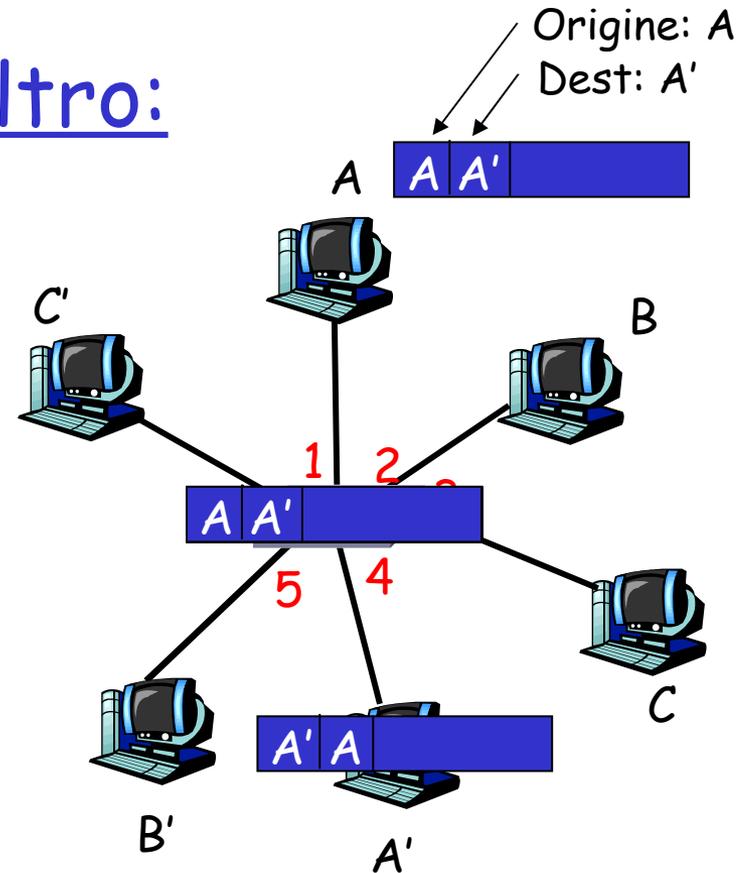
**else** flood

*Lo inoltra a tutti tranne all'interfaccia dalla quale è arrivato il pacchetto*

# Autoapprendimento, inoltra: un esempio

- ❑ Destinazione del frame ignota: *flood*
- ❑ destinazione A, location nota: *selective send*

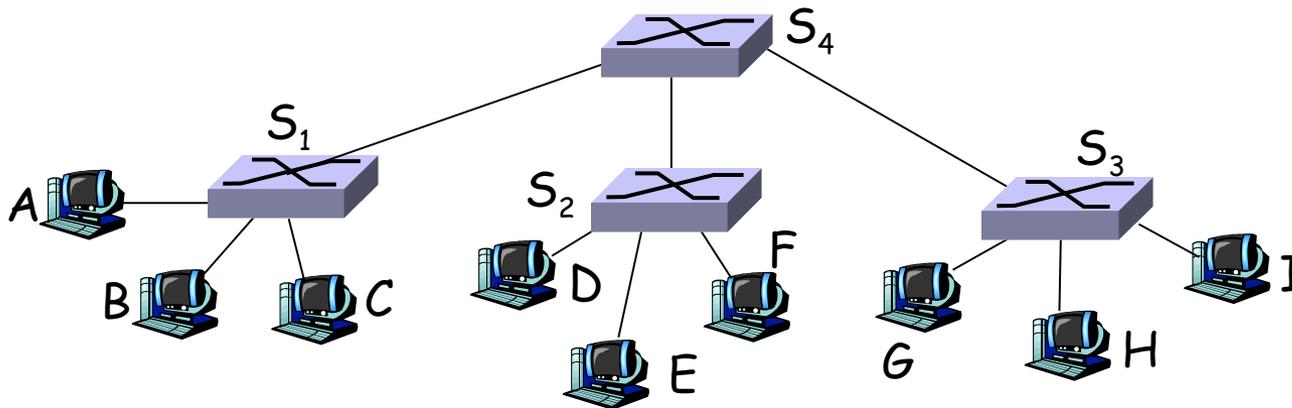
Indir. MAC	Interfaccia	TTL
A	1	60
A'	4	60



*Tabella di commutazione  
(inizialmente vuota)*

# Collegare gli switch

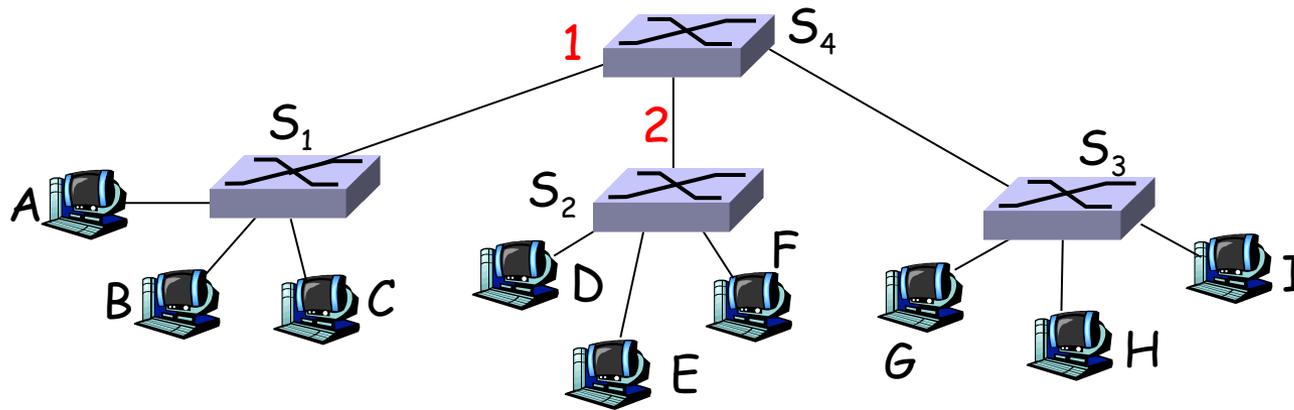
- Gli switch possono essere interconnessi



- D: per inviare da A a G, come fa S<sub>1</sub> a sapere che deve inoltrare il frame attraverso S<sub>4</sub> e S<sub>3</sub>?
- A: autoapprende! (funziona esattamente come nel caso di un singolo switch!)

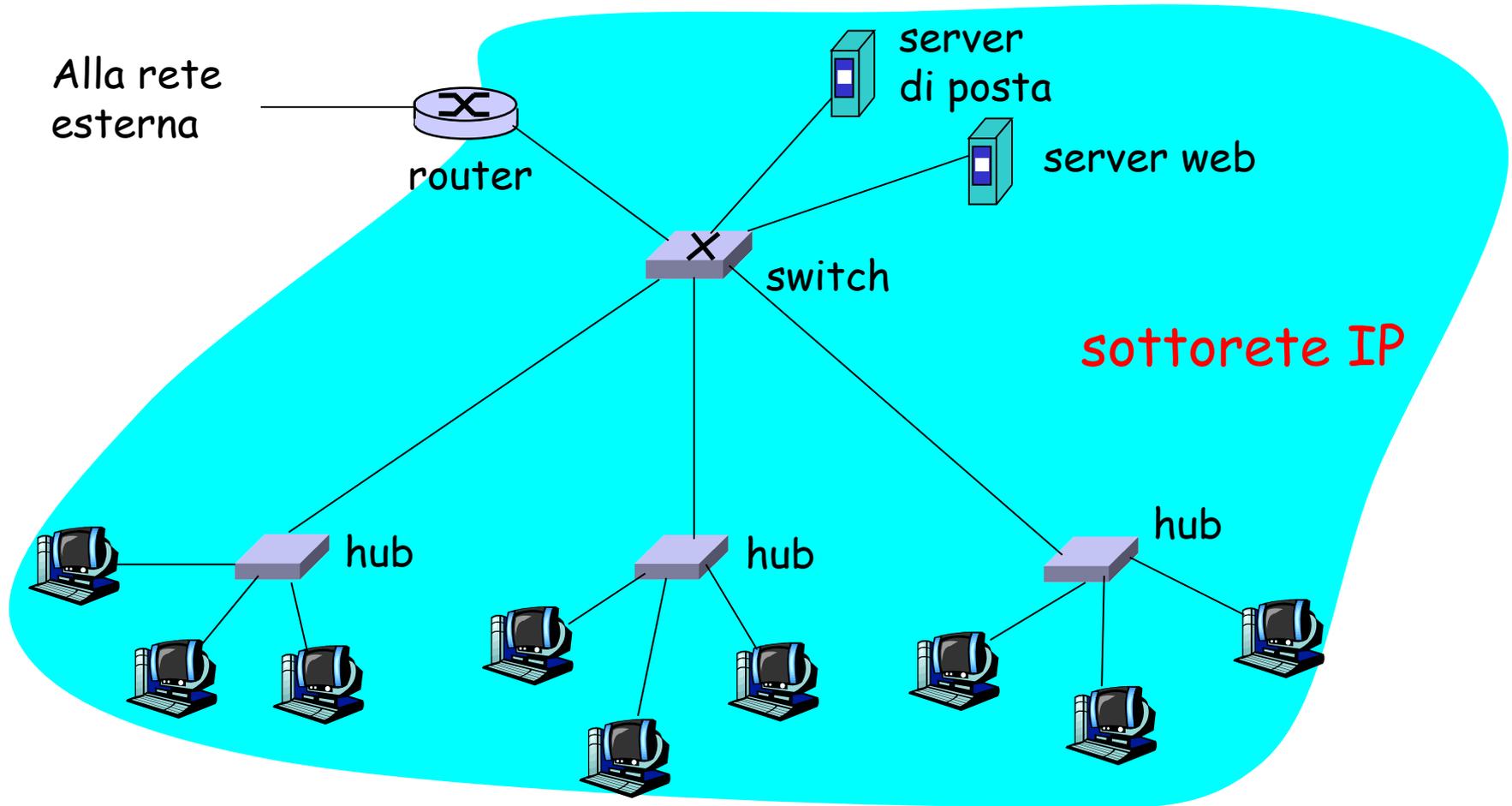
# Esempio di apprendimento multi-switch

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



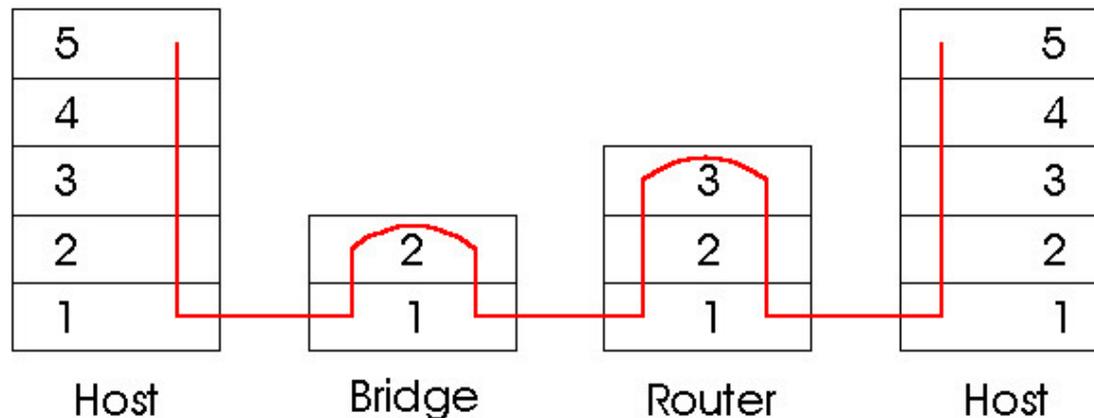
D: illustrate le tabelle di commutazione e l'inoltro dei pacchetti in S<sub>1</sub>, S<sub>2</sub>, S<sub>3</sub>, S<sub>4</sub>

# Esempio di rete di un'istituzione



# Switch e router a confronto

- Entrambi sono dispositivi store-and-forward
  - router: dispositivi a livello di rete (esaminano le intestazioni del livello di rete)
  - switch: dispositivi a livello di collegamento
- I router mantengono tabelle d'inoltro e implementano algoritmi d'instradamento
- Gli switch mantengono tabelle di commutazione e implementano il filtraggio e algoritmi di autoapprendimento



# Capitolo 5: Livello di collegamento e reti locali

- 5.1 Livello di collegamento: introduzione e servizi
- 5.2 Tecniche di rilevazione e correzione degli errori
- 5.3 Protocolli di accesso multiplo
- 5.4 Indirizzi a livello di collegamento
- 5.5 Ethernet
- 5.6 Switch a livello di collegamento
- 5.7 PPP: protocollo punto-punto
- 5.8 Canali virtuali: una rete come un livello di link

# Protocollo punto-punto

- Un mittente, un destinatario, un collegamento: estremamente semplice.
  - non c'è protocollo di accesso al mezzo (MAC)
  - non occorre indirizzamento MAC esplicito
  - il collegamento potrebbe essere una linea telefonica serale commutata, un collegamento SONET/SDH, una connessione X.25 o un circuito ISDN
- Protocolli punto-punto DLC più diffusi:
  - PPP (*point-to-point protocol*)
  - HDLC (*high-level data link control*)

# Requisiti di IETF per il progetto PPP [RFC 1547]

- ❑ **Framing dei pacchetti:** il protocollo PP del mittente incapsula un pacchetto a livello di rete all'interno del un pacchetto PPP a livello di link.
- ❑ **Trasparenza:** il protocollo PPP non deve porre alcuna restrizione ai dati che appaiono nel pacchetto a livello di rete.
- ❑ **Rilevazione degli errori** (ma non la correzione)
- ❑ **Disponibilità della connessione:** il protocollo deve rilevare la presenza di eventuali guasti a livello di link e segnalare l'errore al livello di rete.
- ❑ **Negoziazione degli indirizzi di rete:** PPP deve fornire un meccanismo ai livelli di rete comunicanti per ottenere o configurare gli indirizzi di rete.

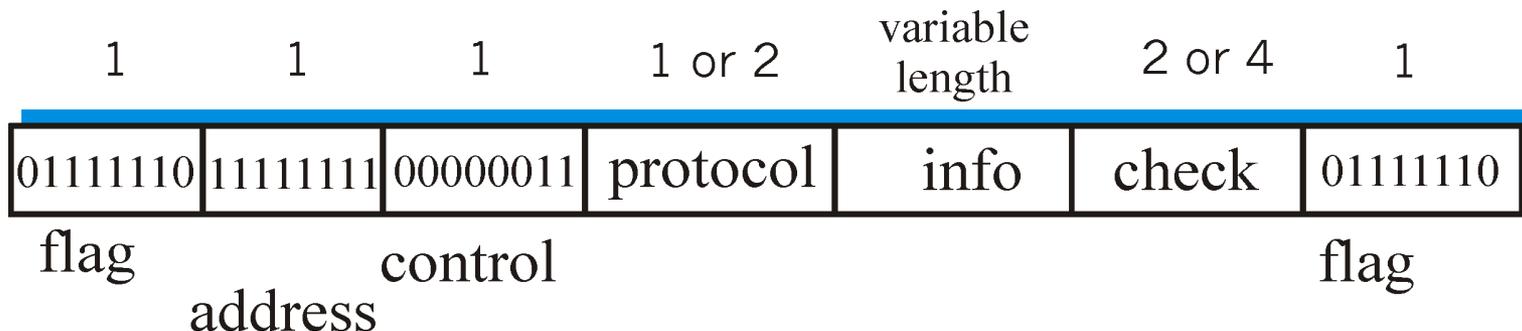
# Requisiti che PPP non deve implementare

- ❑ Correzione degli errori.
- ❑ Controllo di flusso.
- ❑ Sequenza (non deve necessariamente trasferire i pacchetti al ricevente mantenendo lo stesso ordine).
- ❑ Collegamento multipunto (es., polling).

Correzione degli errori, controllo di flusso,  
ri-ordinamento dei pacchetti  
sono delegati ai livelli superiori!

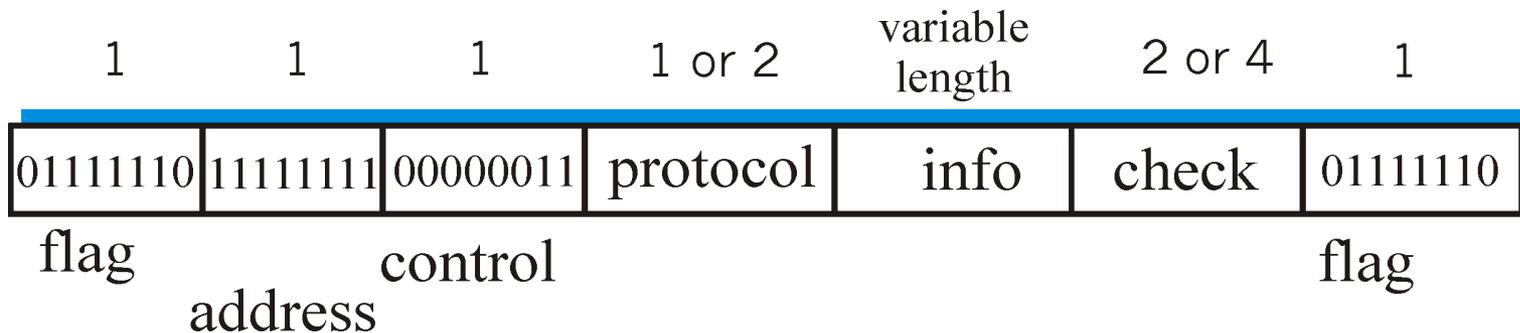
# Formato dei pacchetti dati PPP

- ❑ **Flag:** ogni pacchetto inizia e termina con un byte con valore 01111110
- ❑ **Indirizzo:** unico valore (11111111)
- ❑ **Controllo:** unico valore; ulteriori valori potrebbero essere stabiliti in futuro
- ❑ **Protocollo:** indica al PPP del ricevente qual è il protocollo del livello superiore cui appartengono i dati incapsulati



# Formato dei pacchetti dati PPP

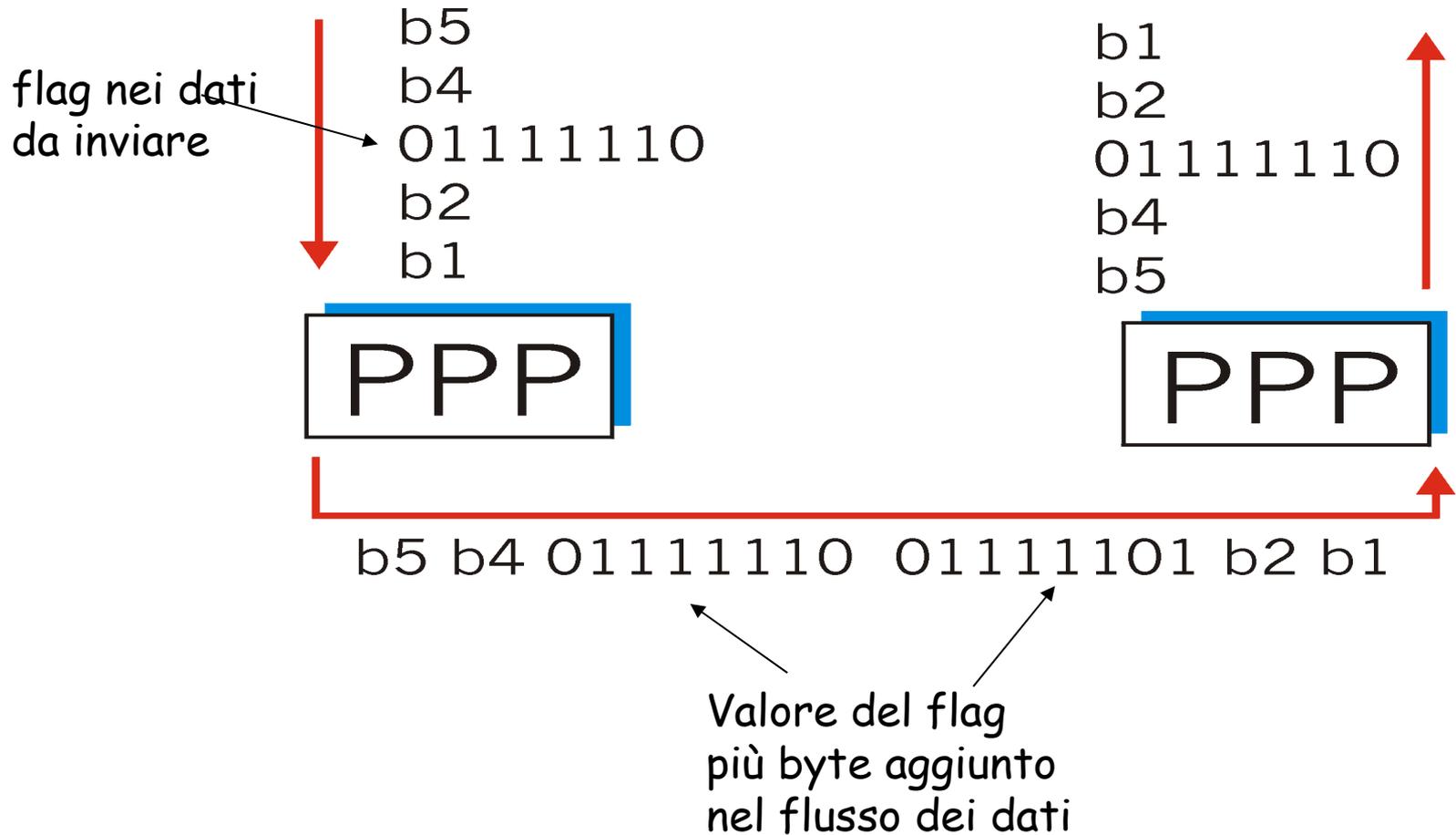
- **informazioni:** incapsula il pacchetto trasmesso da un protocollo del livello superiore (come IP) sul collegamento PPP.
- **checksum:** utilizzato per rilevare gli errori nei bit contenuti in un pacchetto; utilizza un codice a ridondanza ciclica HDLC a due o a quattro byte.



# Riempimento dei byte (Byte stuffing)

- ❑ Requisito di trasparenza: nel campo informazioni deve essere possibile inserire una stringa <01111110>
  - D: se compare <01111110> come fa il ricevente a rilevare in modo corretto la fine del frame PPP?
- ❑ **Mittente**: aggiunge ("stuff") un byte di controllo
  - ❑ <01111110> prima di ogni byte di dati <01111110>
- ❑ **Destinatario**:
  - Due byte 01111110: scarta il primo e continua la ricezione dei dati.
  - Singolo 01111110: valore di flag

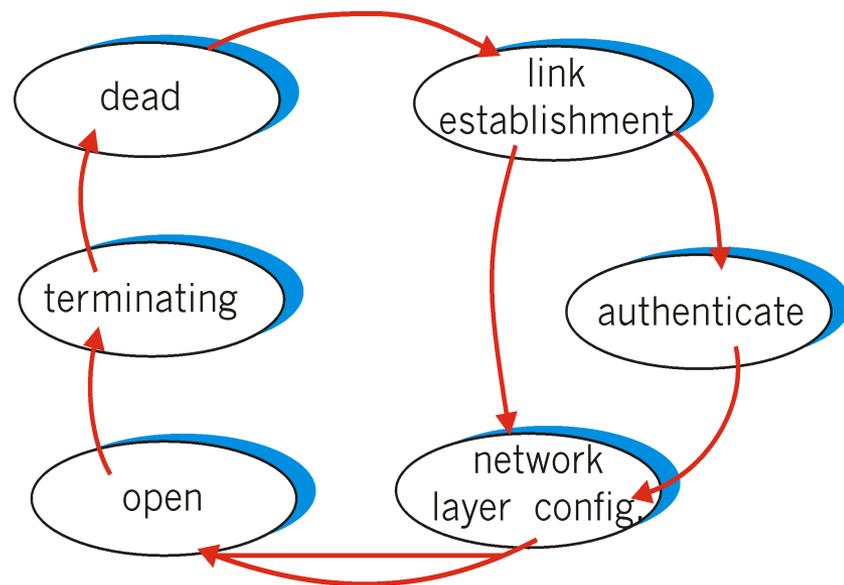
# Byte stuffing



# Protocollo di controllo del collegamento e protocolli di rete

Prima di avviare lo scambio di dati, i due peer devono configurare il collegamento:

- **Configurazione del collegamento PPP** (massima dimensione del pacchetto, autenticazione).
- **Scambio dei pacchetti di controllo propri del livello di rete**
  - per IP: viene utilizzato il protocollo di controllo IP (IPCP) e i dati IPCP sono inseriti in un pacchetto PPP (il cui campo protocollo contiene 8021) in modo analogo a quello in cui i dati LCP sono inseriti in un pacchetto PPP



# Capitolo 5: Livello di collegamento e reti locali

- 5.1 Livello di collegamento: introduzione e servizi
- 5.2 Tecniche di rilevazione e correzione degli errori
- 5.3 Protocolli di accesso multiplo
- 5.4 Indirizzi a livello di collegamento
- 5.5 Ethernet
- 5.6 Switch a livello di collegamento
- 5.7 PPP: protocollo punto-punto
- 5.8 Canali virtuali: una rete come un livello di link

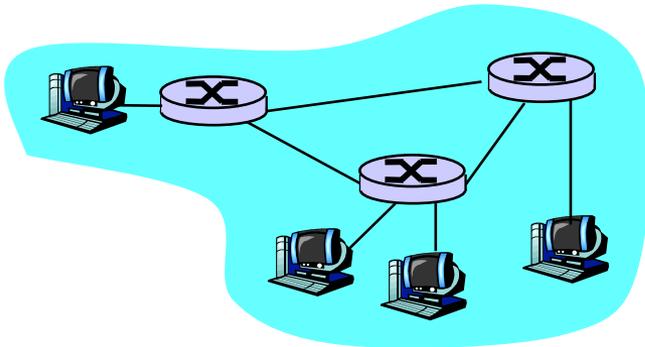
# Virtualizzazione delle reti

- ❑ Virtualizzazione delle risorse: una potente astrazione nell'ingegneria dei sistemi
- ❑ Esempi nell'informatica: memoria virtuale, dispositivi virtuali
  - macchine virtuali: es. Java
  - Sistema operativo IBM VM tra gli anni '60 e '70
- ❑ Il collegamento va considerato in modo astratto, come un canale piuttosto che un cavo

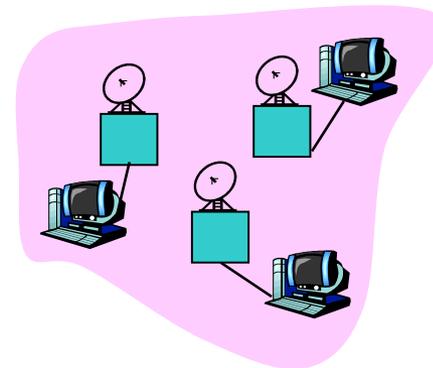
# Internet: virtualizzazione delle reti

1974: tante reti non collegate ... che differiscono per:  
tra loro

- ARPAnet
- Reti su cavo
- Reti a pacchetto satellitari (Aloha)
- Reti radio
- convenzioni di indirizzamento
- formato dei pacchetti
- ripristino degli errori
- instradamento



ARPAnet



rete satellitare

# Internet: virtualizzazione delle reti

## IP:

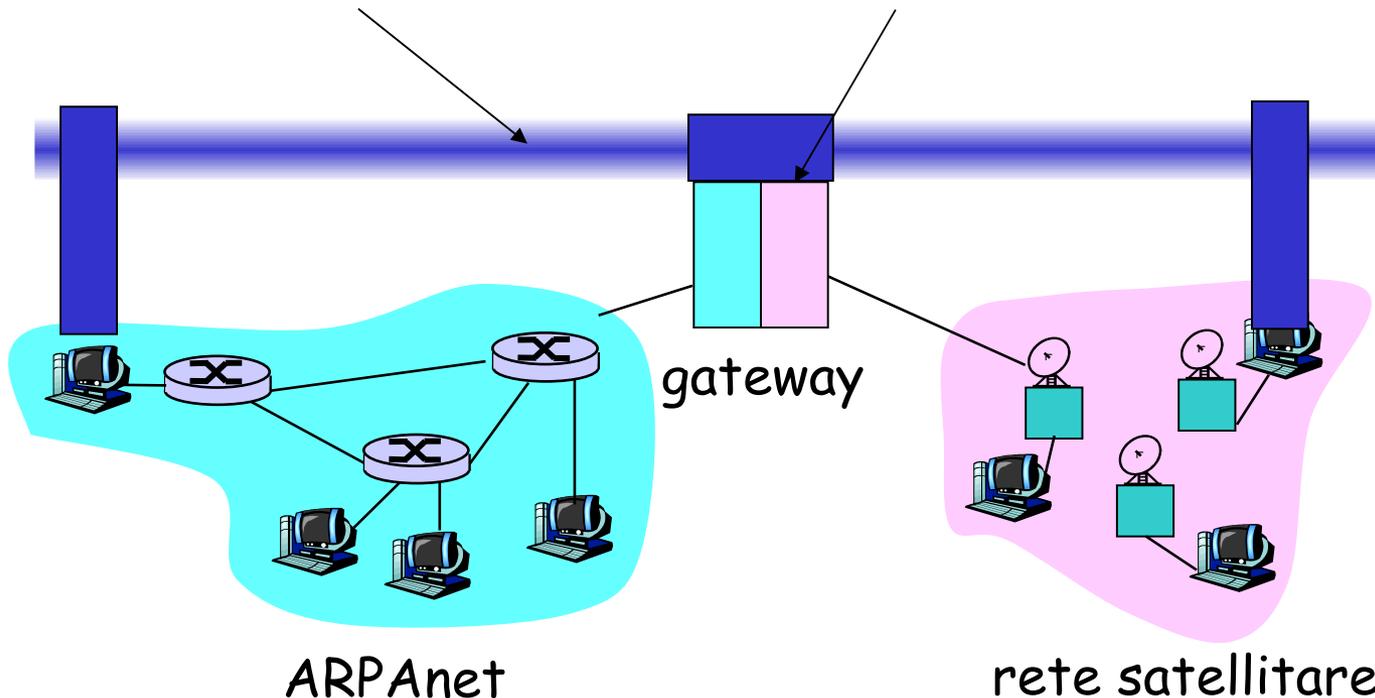
### □ Indirizzamento:

l'interconnessione delle reti fa sì che tutto appaia come una singola entità uniforme nonostante la sottostante eterogeneità delle reti locali

### □ Rete delle reti

## Gateway:

- "racchiude i pacchetti della rete globale in pacchetti con il formato della rete locale o li estrae da questi"
- Li instrada verso il gateway successivo



# L'architettura di Cerf & Kahn

Che cosa è virtualizzato?

- Due livelli di indirizzamento: rete globale e rete locale
- Il nuovo livello (IP) rende tutto omogeneo
- Tecnologia sottostante della rete locale
  - cavo
  - satellite
  - modem telefonico a 56K
  - oggi: ATM, MPLS

# 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
  - LAN commutate
  - PPP
  - Reti virtuali: ATM, MPLS

# Capitolo 5: facciamo il punto

- ❑ Abbiamo **completato il nostro viaggio** attraverso la pila dei protocolli (ad eccezione del livello fisico)
- ❑ Abbiamo una **solida conoscenza dei principi del networking**, e anche degli aspetti pratici
- ❑ Potremmo fermarci qui... ma ci sono ancora **un sacco di argomenti interessanti!**
  - Wireless
  - Multimedia
  - Sicurezza di rete
  - Gestione della rete