

Livello di Rete:  
Router, formato dei datagrammi,  
frammentazione

Gaia Maselli  
maselli@di.uniroma1.it

# Livello di rete

Routing, forwarding,  
gestione connessione

Reti a circuito virtuale  
e a datagramma

Che cosa si trova  
all'interno di un  
router?

Protocollo Internet (IP)

- Formato dei datagrammi
- Indirizzamento IPv4
- ICMP
- IPv6

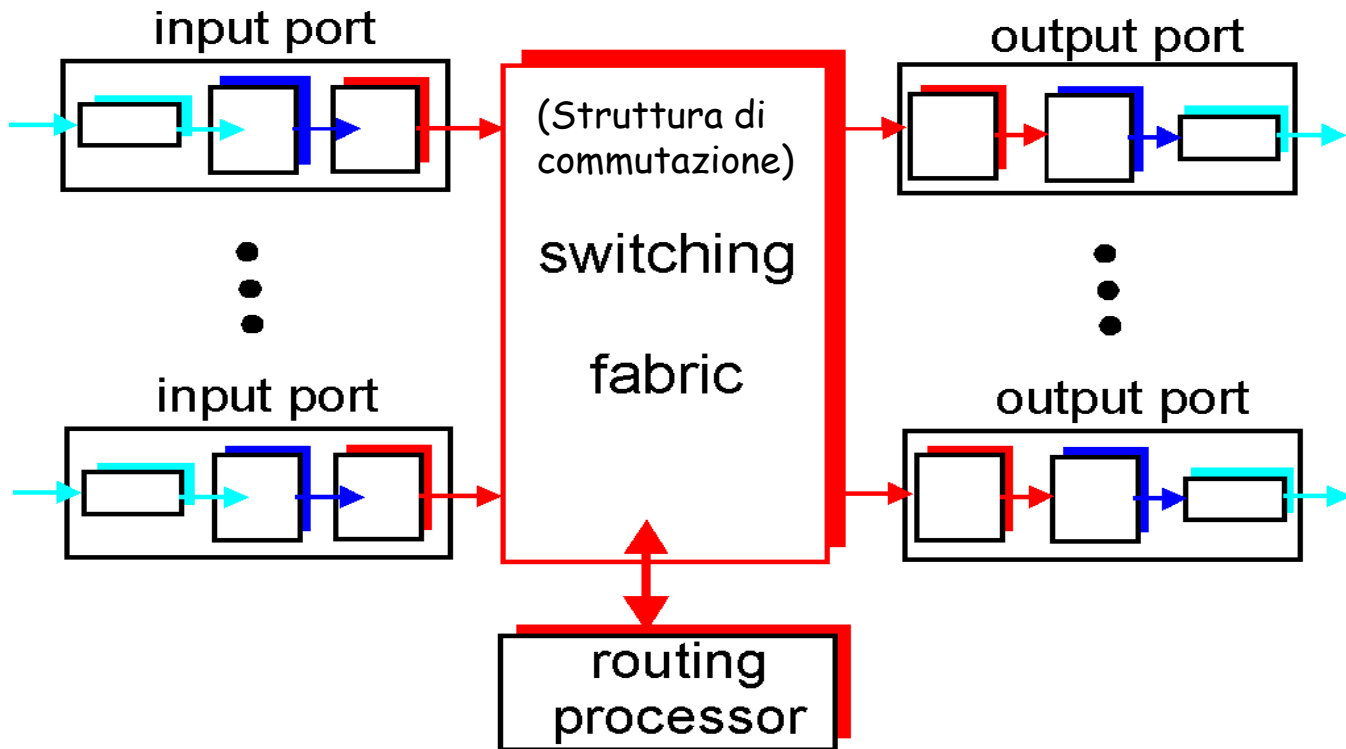
# Livello di rete

- ❑ Routing, forwarding, instaurazione connessione
- ❑ Reti a circuito virtuale
  - Decisione di forwarding si basa sul numero di circuito
- ❑ Reti a datagramma (Internet)
  - Decisione di forwarding si basa sull'indirizzo IP di destinazione
- ❑ **Che cosa si trova all'interno di un router?**
- ❑ Protocollo Internet (IP)
  - Formato dei datagrammi
  - frammentazione

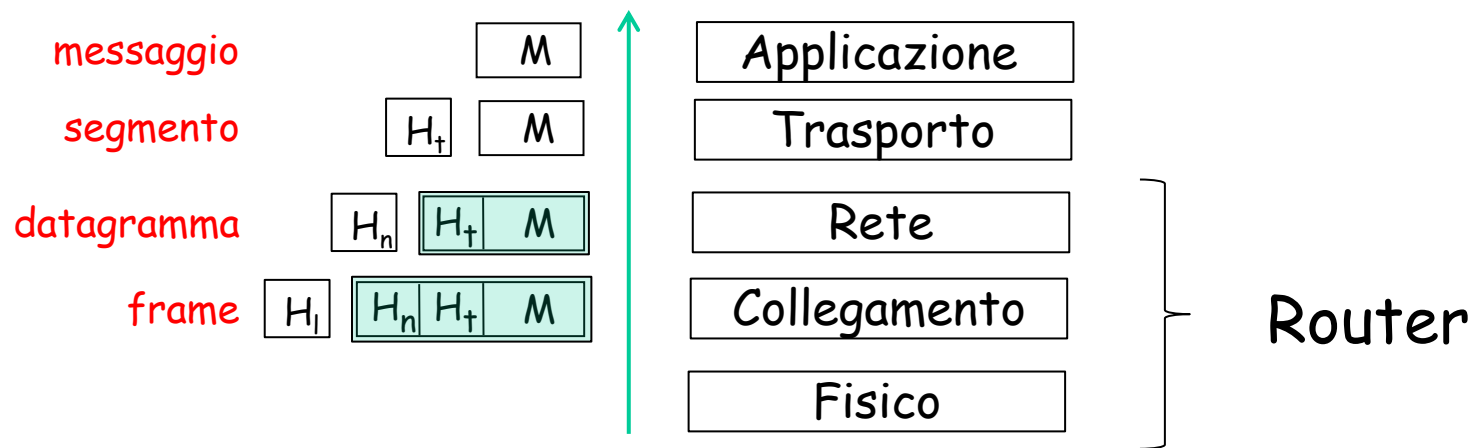
# Architettura del router

Due funzioni chiave:

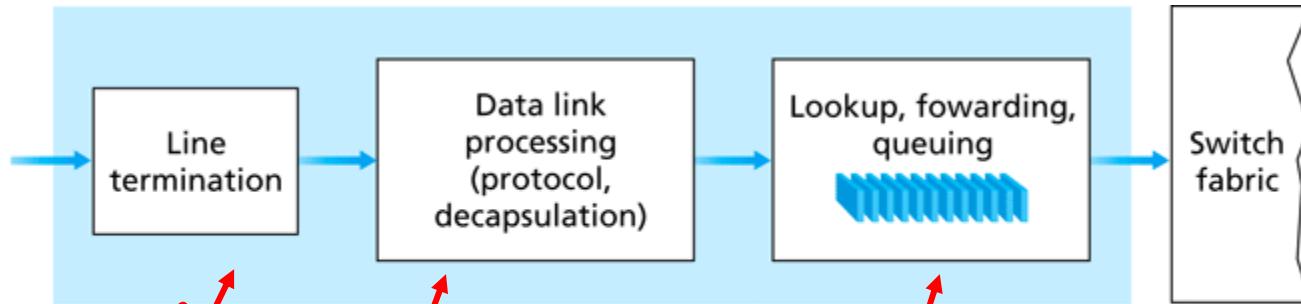
- ❑ Far girare i protocolli/algoritmi d'instradamento (RIP, OSPF, BGP)
- ❑ *Inoltro* di datagrammi dai collegamenti in ingresso a quelli in uscita.



# Funzioni del router



# Porte d'ingresso



**Livello fisico:**  
ricezione a livello di bit  
(terminazione elettrica)

**Livello di collegamento:**  
Es. Ethernet

## **Commutazione decentralizzata:**

- ❑ Determina la porta d'uscita dei pacchetti utilizzando le informazioni della tabella d'inoltro (c'è una copia della tabella memorizzata nella porta di ingresso)
- ❑ Obiettivo: completare l'elaborazione allo stesso **tasso della linea** (evitare colli di bottiglia)
- ❑ Accodamento: se il tasso di arrivo dei datagrammi è superiore a quello di inoltro
- ❑ Una volta determinata la porta di uscita il pacchetto verrà inoltrato alla struttura di commutazione

# Porte di ingresso in una workstation

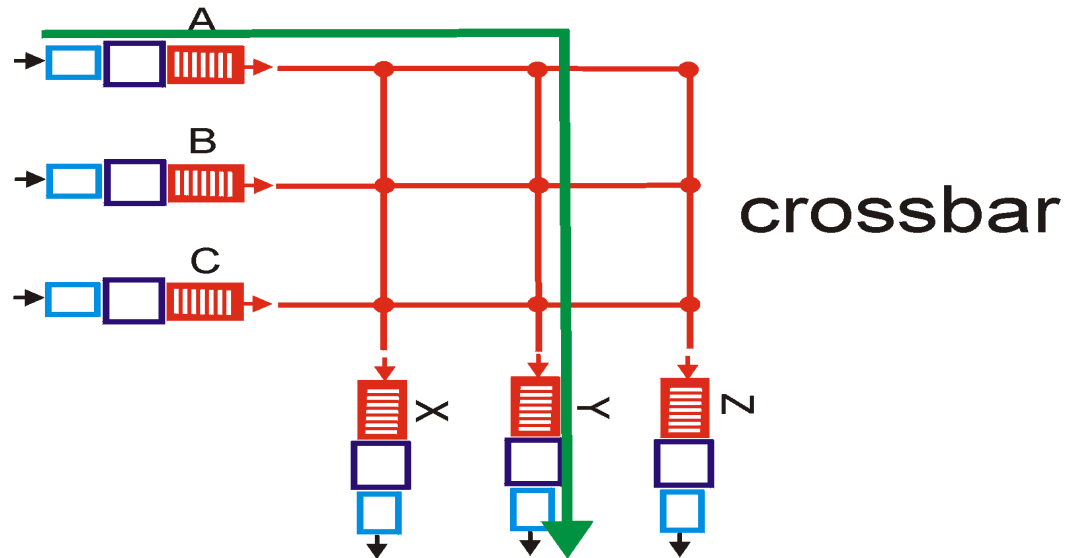
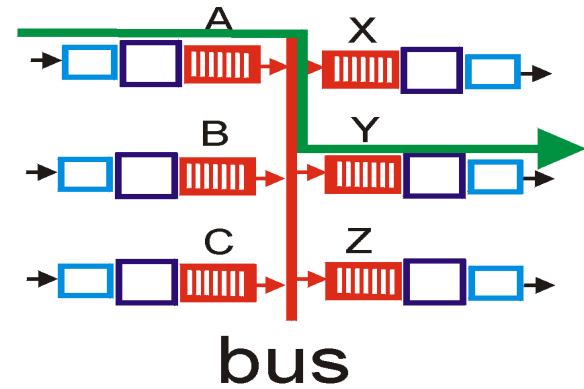
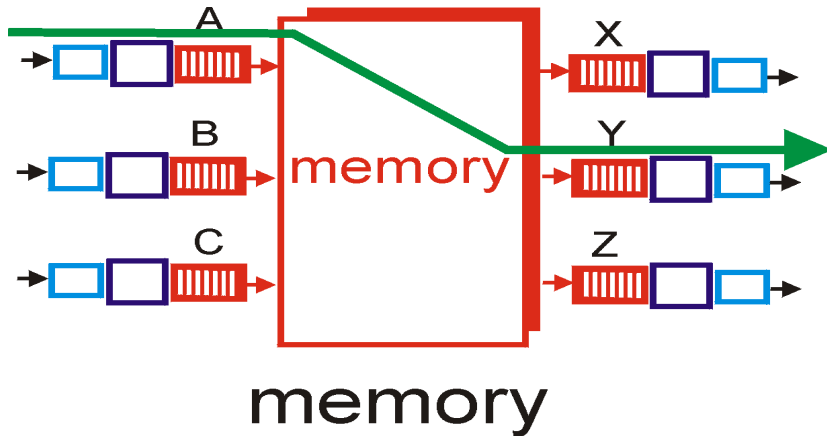
- Nel caso di workstation o server che funzionano da router (quindi dispositivi non dedicati al routing)
  - il processore di instradamento è proprio la CPU della workstation
  - la porta di ingresso coincide con la scheda di rete (es. Ethernet)

# Ricerca nella tabella di inoltra

- ❑ Deve essere veloce (possibilmente stesso tasso della linea) per evitare accodamenti
  - ❑ Tabella implementata in una **struttura ad albero**
    - Ogni livello dell'albero corrisponde a un bit dell'indirizzo di destinazione
    - Per cercare un indirizzo si comincia dalla radice dell'albero
      - Se 0 allora sottoalbero di sinistra
      - Se 1 allora sottoalbero di destra
    - Ricerca in N passi dove N è il numero di bit nell'indirizzo
  - ❑ Argomento molto studiato, per approfondiment
- Survey and taxonomy of IP address lookup algorithms, IEEE Newtork 2001**



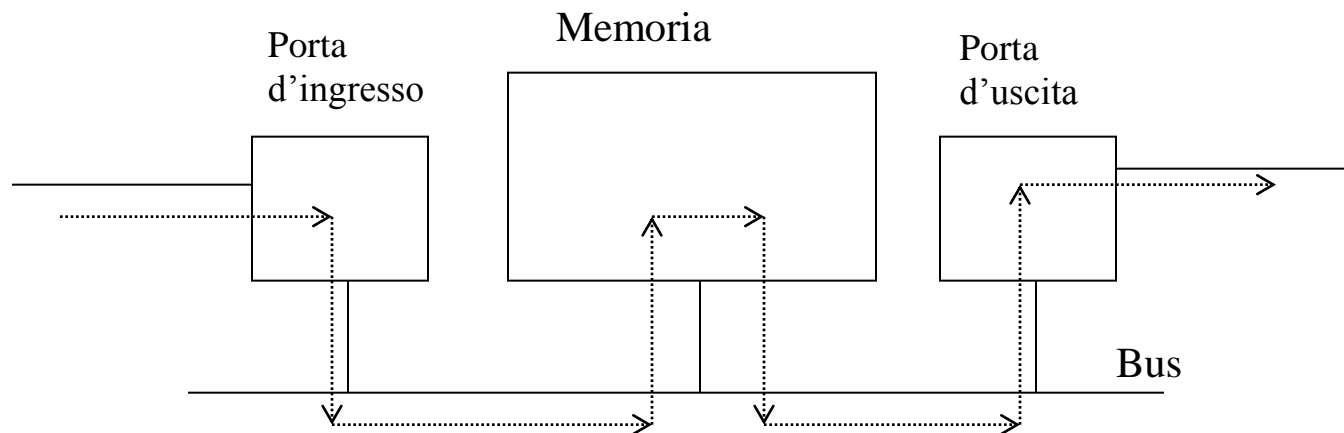
# Tre tecniche di commutazione



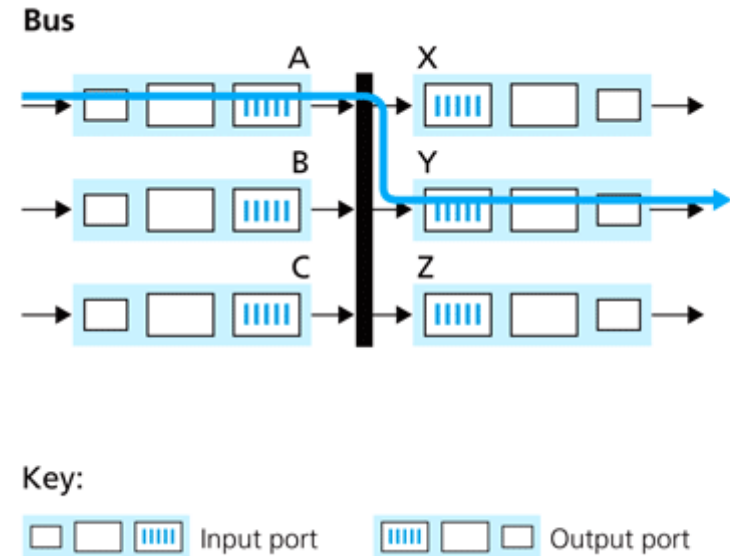
# Commutazione in memoria

## Prima generazione di router:

- ❑ Erano tradizionali calcolatori e la commutazione era effettuata sotto il controllo diretto della CPU.
- ❑ Il pacchetto veniva copiato nella memoria del processore.
- ❑ I pacchetti venivano trasferiti dalle porte d'ingresso a quelle d'uscita con una frequenza totale inferiore a  $B/2$ , dove  $B$  è il numero di pacchetti che si possono scrivere o leggere in memoria

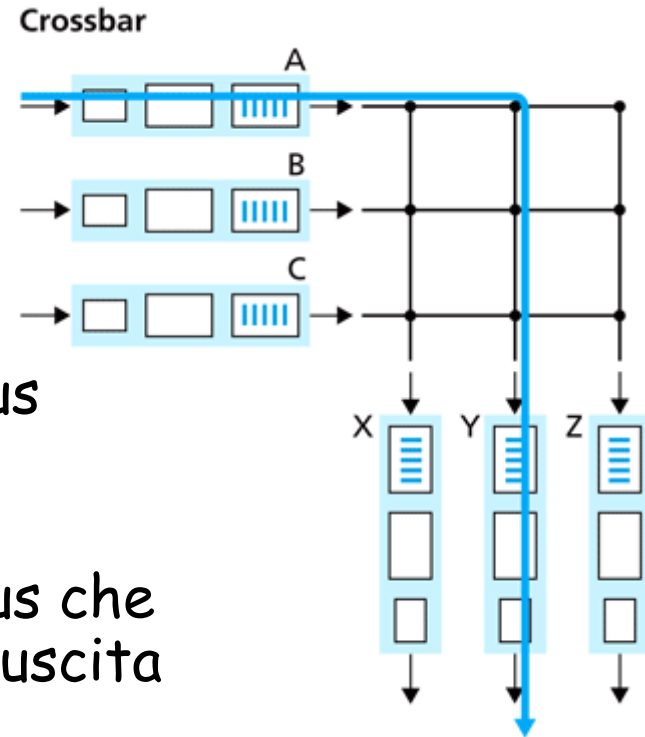


# Commutazione tramite bus



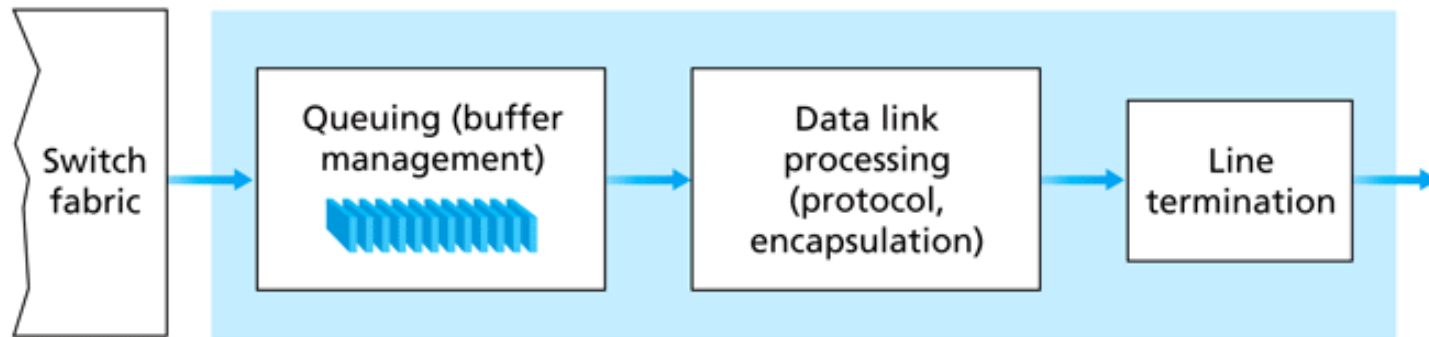
- ❑ Le porte d'ingresso trasferiscono un pacchetto direttamente alle porte d'uscita su un bus condiviso, senza intervento del processore di instradamento.
- ❑ Si può trasferire un solo pacchetto alla volta
- ❑ I pacchetti che arrivano e trovano il bus occupato vengono accodati alla porta di ingresso
- ❑ **Contesa per il bus:** la larghezza di banda della commutazione è limitata da quella del bus.
- ❑ Cisco 5600 opera con bus da 32 Gbps: è sufficiente per router che operano in reti d'accesso o in quelle aziendali

# Commutazione attraverso rete d'interconnessione



- ❑ Supera il limite di banda di un singolo bus condiviso.
- ❑ Un **crossbar switch** è una rete d'interconnessione che consiste di  $2n$  bus che collegano  $n$  porte d'ingresso a  $n$  porte d'uscita
- ❑ Tendenza attuale: frammentazione dei pacchetti IP a lunghezza variabile in celle di lunghezza fissa (riasmblati nella porta di uscita).
- ❑ Switch Cisco 12000: usano una rete d'interconnessione che raggiunge i 60 Gbps nella struttura di commutazione.

# Porte d'uscita



**Figure 4.9** ♦ Output port processing

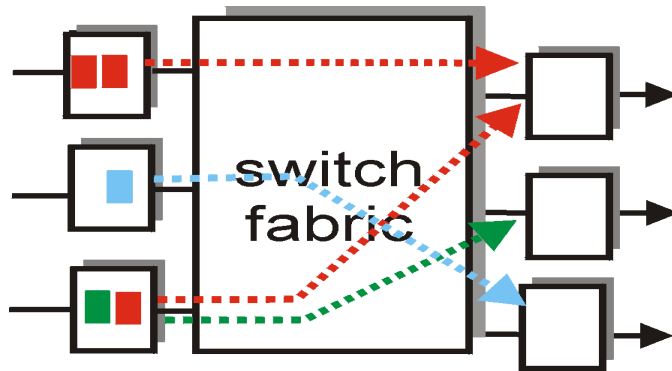
- ❑ **Funzionalità di accodamento:** quando la struttura di commutazione consegna pacchetti alla porta d'uscita a una frequenza che supera quella del collegamento uscente.
- ❑ **Schedulatore di pacchetti:** stabilisce in quale ordine trasmettere i pacchetti accodati.

# Dove si verifica l'accodamento?

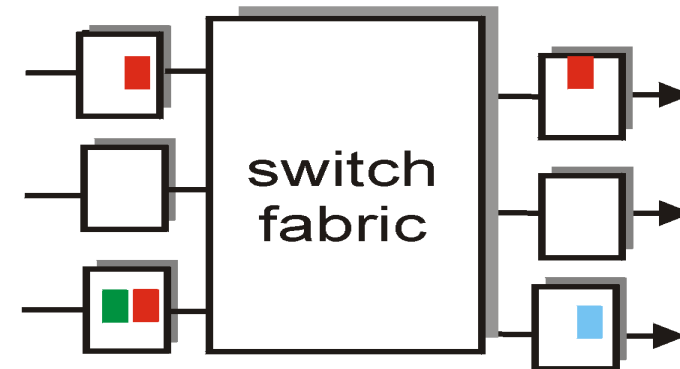
- Sia nelle porte di **ingresso** che nelle porte di **uscita**
  - Velocità di commutazione: frequenza alla quale tale struttura può trasferire i pacchetti dalle porte di ingresso a quelle di uscita
  - **Accodamento nelle porte di ingresso** - quando la struttura di commutazione ha una velocità inferiore a quello delle porte di ingresso (per non avere accodamento la velocità di commutazione dovrebbe essere  $n \times$  velocità della linea di ingresso)
  - **Accodamento nelle porte di uscita**
    - Quando la struttura di commutazione ha un rate superiore alla porta di uscita
    - Quando troppi pacchetti vanno sulla stessa porta di uscita

# Accodamento su porte di ingresso

- Oltre alla velocità inferiore della struttura di commutazione
- **Blocco in testa alla fila (HOL: head-of-the-line blocking):** un pacchetto nella coda d'ingresso deve attendere il trasferimento (anche se la propria destinazione è libera) in quanto risulta bloccato da un altro pacchetto in testa alla fila.
- **Se le code diventano troppo lunghe, i buffer si possono saturare e quindi causare una perdita di pacchetti!**



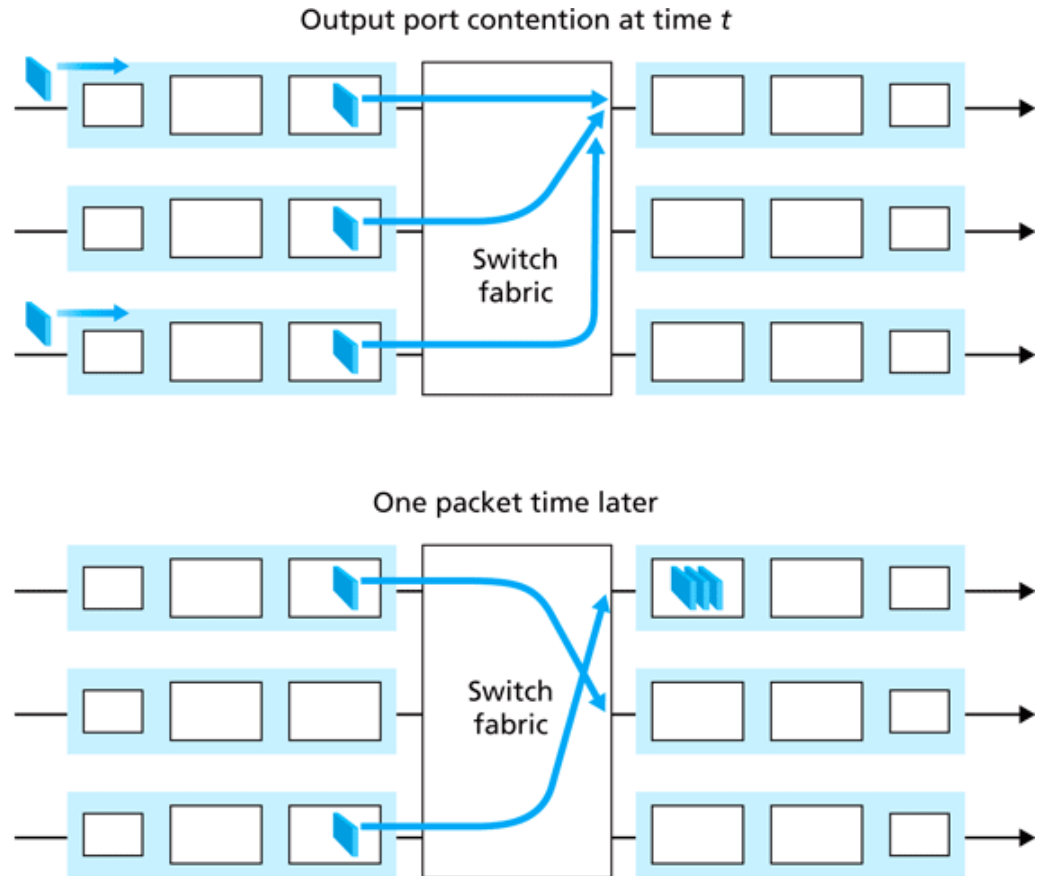
output port contention  
at time t - only one red  
packet can be transferred



green packet  
experiences HOL blocking

# Accodamento sulle porte di uscita

- Se la struttura di commutazione non è sufficientemente rapida nel trasferire i pacchetti, si può verificare un accodamento.
- Se troppi pacchetti vanno sulla stessa uscita
- Se le code diventano troppo lunghe, i buffer si possono saturare e quindi causare una perdita di pacchetti!





# Quale deve essere la capacità dei buffer?

- ❑ La regola spannometrica della RFC 3439 diceva che la quantità di buffering dovrebbe essere uguale a una media del tempo di andata e ritorno (RTT ad esempio 250 msec) per la capacità del collegamento  $C$ 
  - Es.:  $C =$  collegamento da 10 Gps: buffer 2.5 Gbit
- ❑ Attuali raccomandazioni dicono che la quantità di buffering necessaria per  $N$  flussi è:

$$\frac{RTT \cdot C}{\sqrt{N}}$$

# Schedulatore di pacchetti

- Quando vi sono più pacchetti accodati sulle porte di uscita lo schedulatore di pacchetti deve stabilire *in quale ordine trasmetterli*
- **First come first served** (FCFS)
- **Accodamento equo ponderato** (weighted fair queuing, WFQ): ripartisce equamente il collegamento uscente tra diverse connessioni che hanno pacchetti accodati per la trasmissione
- Nel caso di coda piena ci sono politiche che stabiliscono se
  - Scartare il pacchetto in arrivo (drop tail)
  - Rimuoverne uno far quelli già in coda
  - Gestione attiva della coda (AQM, active queue management)

# Gestione attiva della coda

- ❑ Politiche per comunicare al mittente un segnale di congestione prima che la coda sia piena
- ❑ **Random Early Detection (RED)**
  - Mantiene una media ponderata della lunghezza della coda in output
  - Due soglie: min e max
  - Lunghezza media  $<$  min  $\rightarrow$  pacchetti ammessi in coda
  - Lunghezza media  $>$  max  $\rightarrow$  pacchetti marcati e scartati
  - Lunghezza  $\in$  [min, max]  $\rightarrow$  pacchetti marcati e eliminati con probabilità dipendente dalla lunghezza della coda

# Livello di rete

Introduzione

Reti a circuito virtuale e  
a datagramma

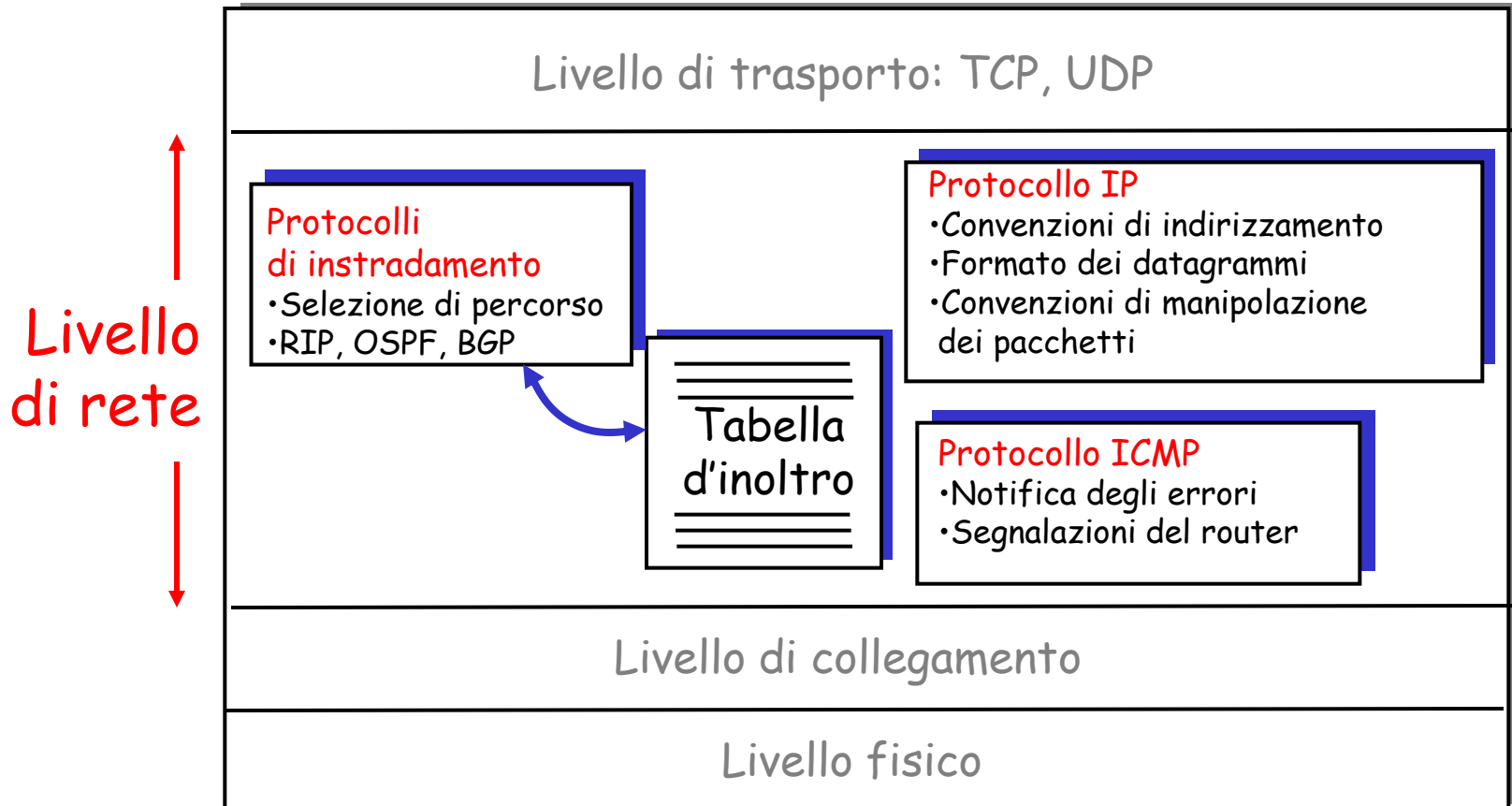
Che cosa si trova  
all'interno di un  
router?

## Protocollo Internet (IP)

- Formato dei datagrammi
- Indirizzamento IPv4
- ICMP
- IPv6

# Protocollo Internet (IP): inoltro e indirizzamento in Internet

Uno sguardo al livello di rete Internet:



# Livello di rete

Introduzione

Reti a circuito virtuale e  
a datagramma

Che cosa si trova  
all'interno di un  
router?

## Protocollo Internet (IP)

- Formato dei datagrammi
- Indirizzamento IPv4
- ICMP
- IPv6

## 4.5 Algoritmi di instradamento

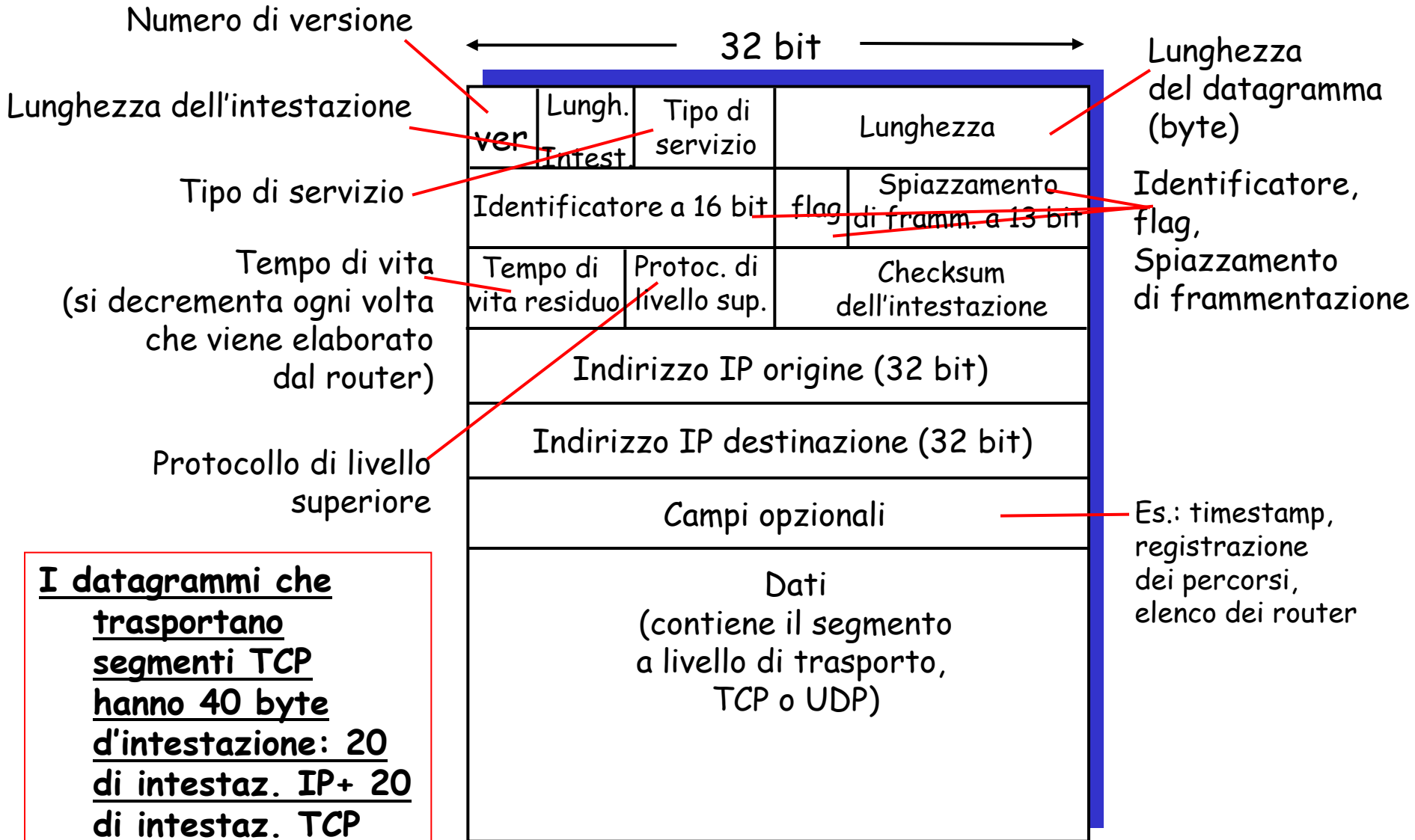
- Stato del collegamento
- Vettore distanza
- Instradamento gerarchico

## 4.6 Instradamento in Internet

- RIP
- OSPF
- BGP

## 4.7 Instradamento broadcast e multicast

# Formato dei datagrammi



**I datagrammi che trasportano segmenti TCP hanno 40 byte d'intestazione: 20 di intestaz. IP+ 20 di intestaz. TCP**

# Formato dei datagrammi

- ❑ **Numero di versione:** consente al router la corretta interpretazione del datagramma
  - IPv4
  - IPv6
- ❑ **Lunghezza dell'intestazione:** poichè un datagramma IP può contenere un numero variabile di opzioni (incluse nell'intestazione), questi bit indicano dove inizia il campo dati. (Intestazione senza opzione = 20 byte)
- ❑ **Tipo di servizio:** serve per distinguere diversi datagrammi con requisiti di qualità del servizio diverse
- ❑ **Lunghezza del datagramma:** rappresenta la lunghezza totale del datagramma IP inclusa l'intestazione (in byte). In genere non superiore ai 1500 byte.



# Formato dei datagrammi

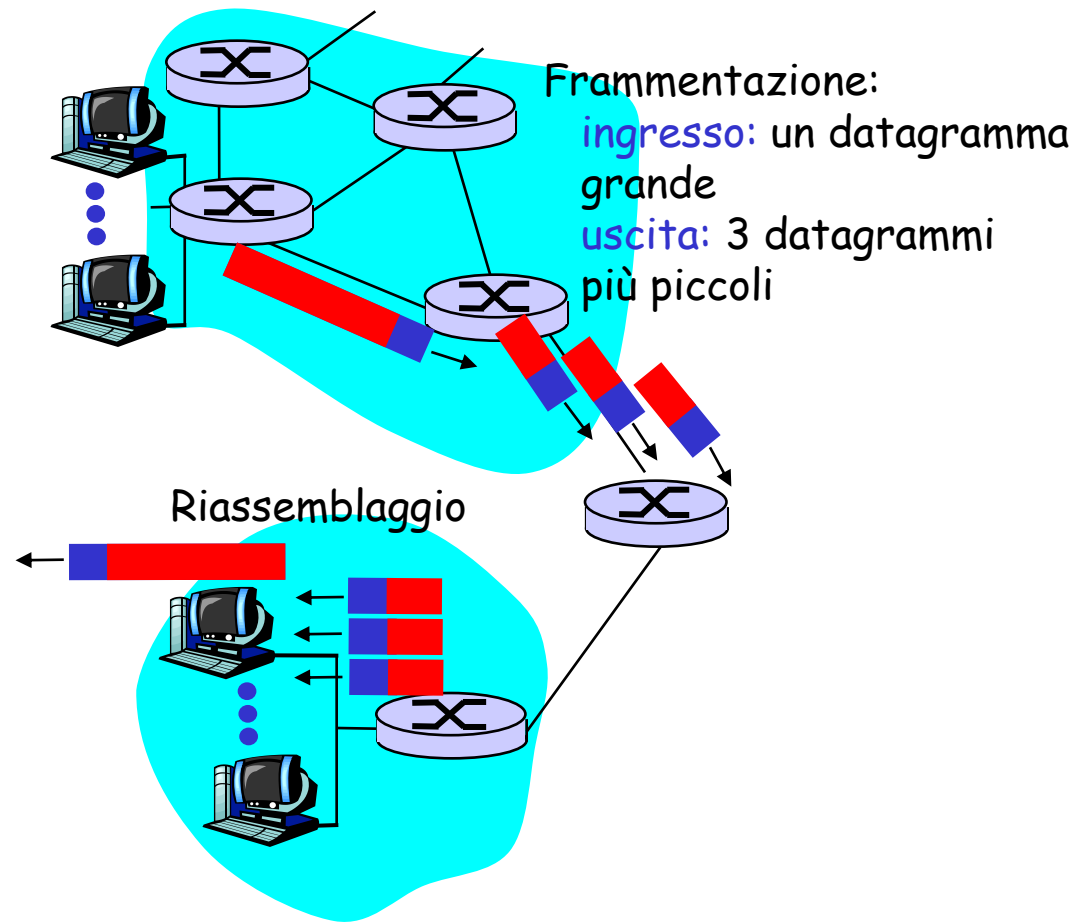
- ❑ **Identificatore, spostamento laterale di frammentazione:** questi tre campi servono per gestire la frammentazione dei pacchetti
- ❑ **Tempo di vita:** o time to live (TTL) è incluso per assicurare che i datagrammi non restino in circolazione per sempre nella rete (in caso per esempio di instradamento ciclico). Il campo viene decrementato a ogni hop e il datagramma viene eliminato in caso il suo  $TTL = 0$
- ❑ **Protocollo:** indica il protocollo a livello di trasporto al quale va passato il datagramma. Questo campo è utilizzato solo quando il datagramma raggiunge la destinazione finale.
  - 6: TCP
  - 17: UDP

# Formato dei datagrammi

- ❑ **Checksum dell'intestazione:** consente ai router di rilevare errori sui datagrammi ricevuti
  - Calcolata solo sull'intestazione
  - Checksum UDP/TCP è calcolata su intero segmento
  - TCP può girare su un protocollo diverso, quale ATM
- ❑ **Indirizzi IP di origine e destinazione:** inseriti dall'host che crea il datagramma (dopo aver effettuato una ricerca DNS)
- ❑ **Opzioni:** campi che consentono di estendere l'intestazione IP
  
- ❑ **Dati:**
  - il campo dati contiene il segmento di trasporto da consegnare alla destinazione
  - può trasportare altri tipi di dati, quali I messaggi ICMP.

# Frammentazione dei datagrammi IP

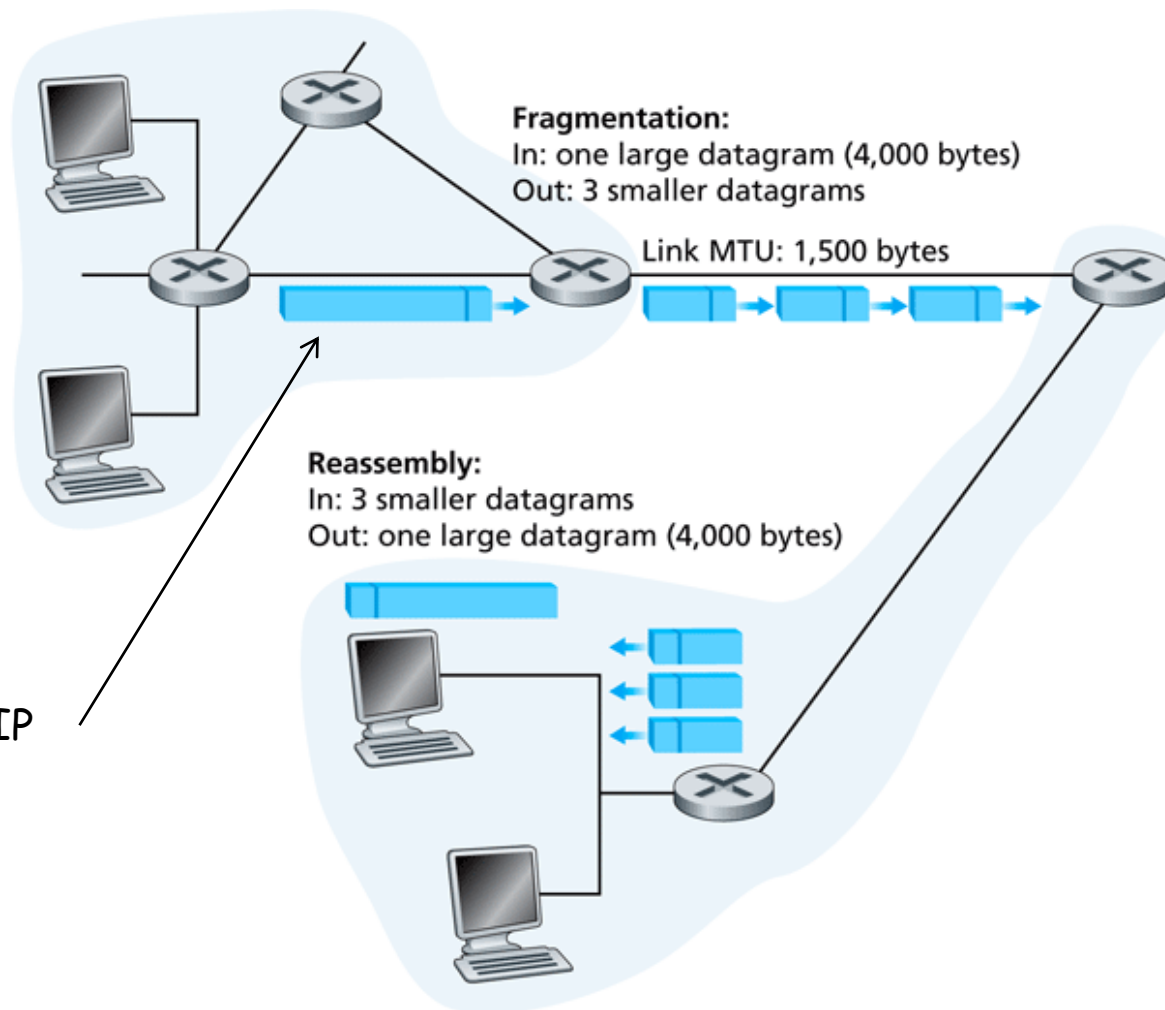
- L'unità massima di trasmissione (MTU) è la massima quantità di dati che un frame a livello di collegamento può trasportare.
  - Differenti tipi di link, differenti MTU.
- Datagrammi IP grandi vengono suddivisi ("frammentati") in datagrammi IP più piccoli.
  - Un datagramma viene frammentato.
  - I frammenti saranno riassemblati solo una volta raggiunta la destinazione
  - I frammenti devono essere riassemblati prima di raggiungere il livello di trasporto
  - I bit dell'intestazione IP sono usati per identificare e ordinare i frammenti



# Bit nell'intestazione per la frammentazione

- ❑ Quando un host di destinazione riceve una serie di datagrammi dalla stessa origine deve
  - individuare i frammenti
  - determinare quando ha ricevuto l'ultimo
  - stabilire come debbano essere riassemblati
- ❑ **Identificazione**: identificativo associato a ciascun datagramma al momento della creazione (unico per tutti i frammenti)
- ❑ **Flag**: indica frammenti intermedi (1) o ultimo frammento (0)
- ❑ **Offset (Spostamento laterale)**: specifica l'ordine del frammento all'interno del datagramma originario

# Frammentazione e riassetblaggio



20 byte intestazione  
3980 byte di carico IP  
ID: 777

# Frammentazione e riassetblaggio

| Fragment     | Bytes  | ID                   | Offset   | Flag   |
|--------------|--|----------------------|--|--|
| 1st fragment | 1,480 bytes in the data field of the IP datagram | identification = 777 | offset = 0 (meaning the data should be inserted beginning at byte 0)   | flag = 1 (meaning there is more)             |
| 2nd fragment | 1,480 bytes of data                              | identification = 777 | offset = 185 (meaning the data should be inserted beginning at byte 1,480. Note that $185 \cdot 8 = 1,480$ ) | flag = 1 (meaning there is more)             |
| 3rd fragment | 1,020 bytes (= $3,980 - 1,480 - 1,480$ ) of data | identification = 777 | offset = 370 (meaning the data should be inserted beginning at byte 2,960. Note that $370 \cdot 8 = 2,960$ ) | flag = 0 (meaning this is the last fragment) |

In totale vengo trasferiti 40 byte in più: ci sono due header aggiuntivi!!!

# Frammentazione e riassetblaggio IP

## Esempio

- Datagramma di 4000 byte
- MTU = 1500 byte

|                   |          |            |               |
|-------------------|----------|------------|---------------|
| Lunghez.<br>=4000 | ID<br>=x | Flag<br>=0 | Spiazz.<br>=0 |
|-------------------|----------|------------|---------------|

Un datagramma IP grande viene frammentato in datagrammi IP più piccoli.

1480 byte nel campo dati

Spiazzamento =  $1480/8$

|                   |          |            |               |
|-------------------|----------|------------|---------------|
| Lunghez.<br>=1500 | ID<br>=x | Flag<br>=1 | Spiazz.<br>=0 |
|-------------------|----------|------------|---------------|

|                   |          |            |                 |
|-------------------|----------|------------|-----------------|
| Lunghez.<br>=1500 | ID<br>=x | Flag<br>=1 | Spiazz.<br>=185 |
|-------------------|----------|------------|-----------------|

|                   |          |            |                 |
|-------------------|----------|------------|-----------------|
| Lunghez.<br>=1040 | ID<br>=x | Flag<br>=0 | Spiazz.<br>=370 |
|-------------------|----------|------------|-----------------|