

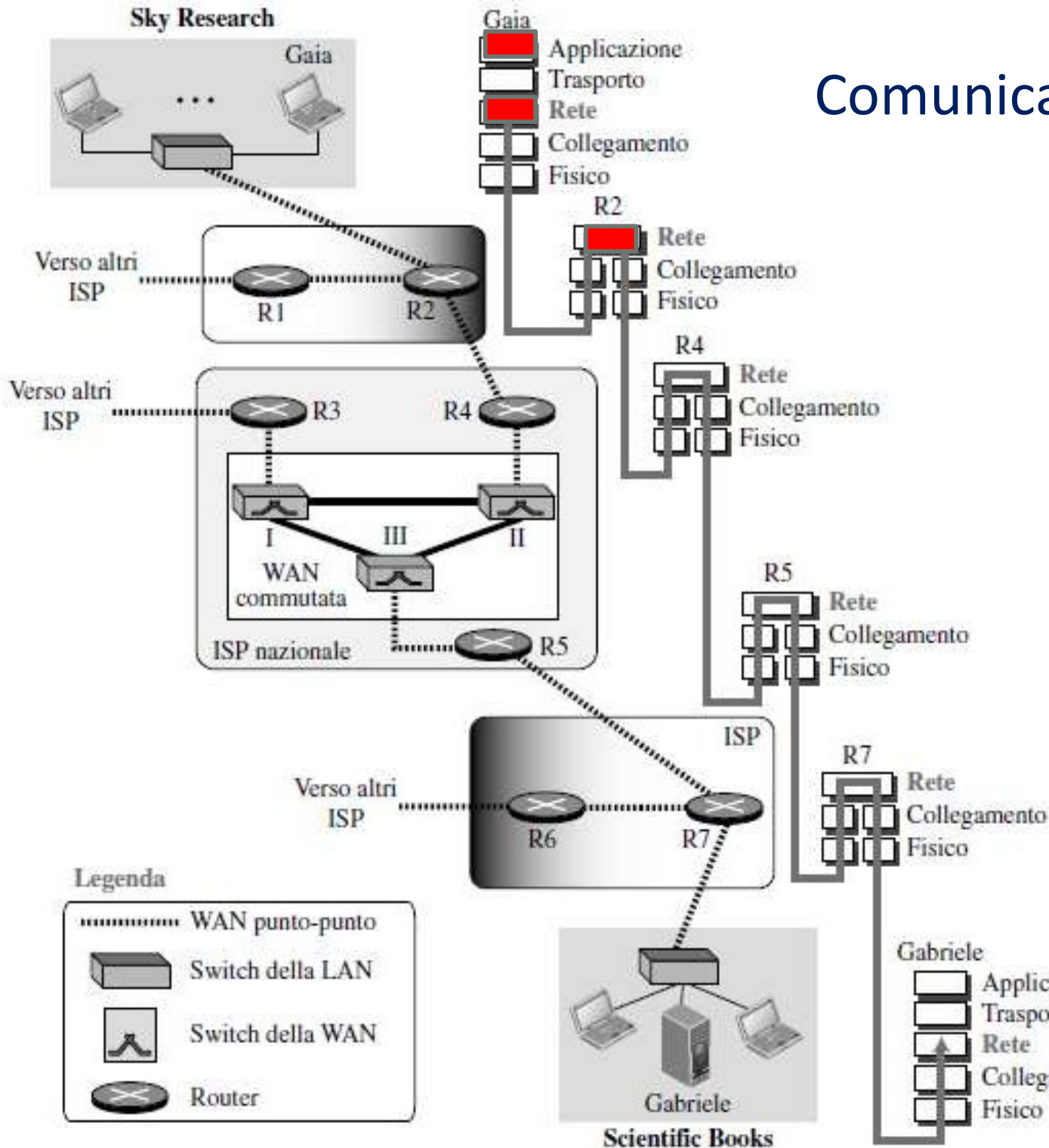
# Lo strato di Rete IP

Federica Paganelli  
2023/24

# Strato di Rete

- Responsabile della consegna dei datagrammi tra gli host
- Offre servizi allo strato di trasporto.
- Utilizza i servizi dello strato collegamento

# Comunicazione a livello rete



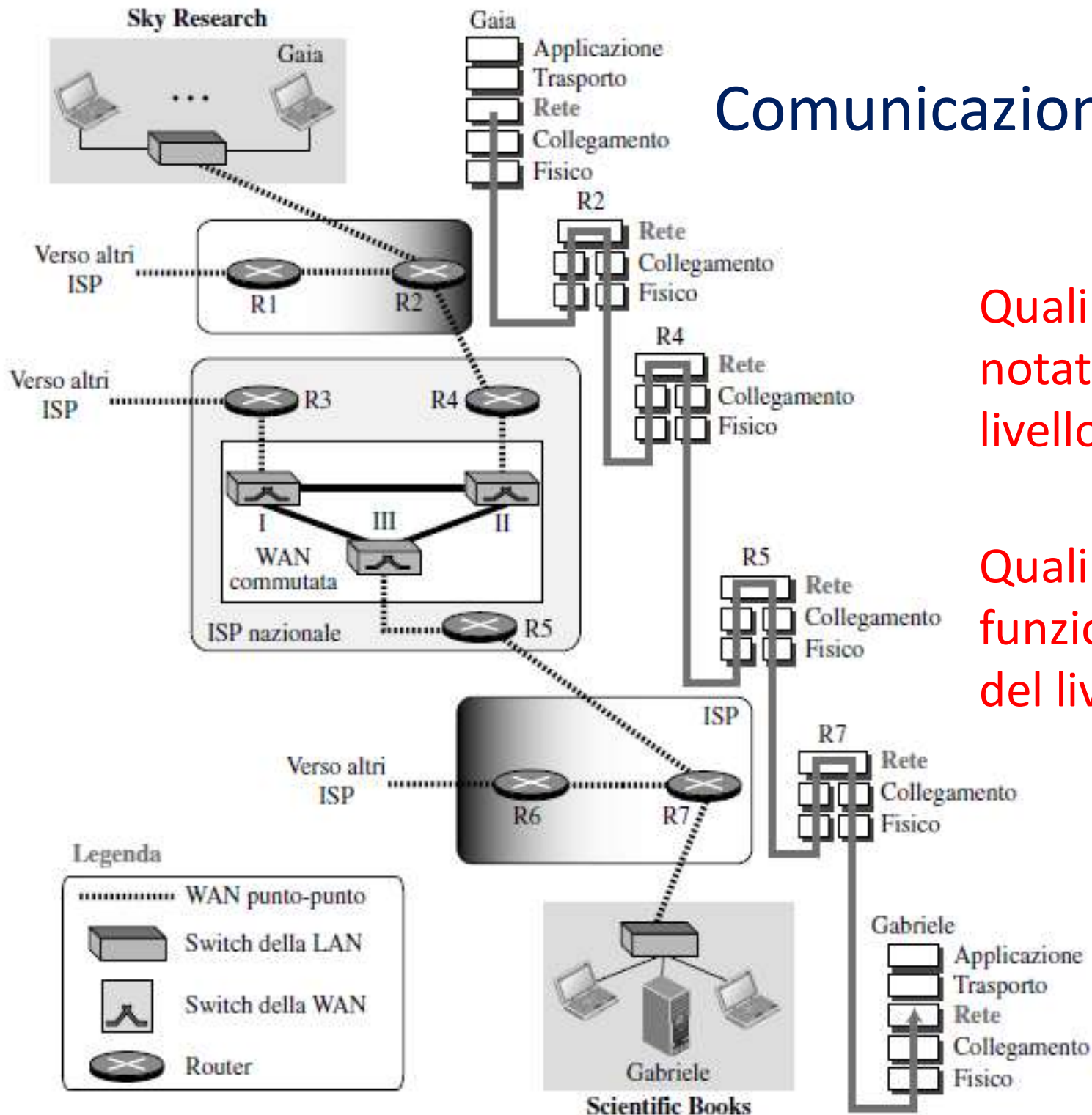
1. L'entità a livello di rete riceve i segmenti dal livello di trasporto nell'host mittente, e incapsula i segmenti in datagrammi

2. I datagrammi sono inoltrati al prossimo nodo (host o router)

3. Il router esamina i campi intestazione in tutti i datagrammi IP che lo attraversano e li inoltra da un collegamento in ingresso ad un collegamento in uscita

4. Sul lato destinatario, consegna i segmenti al livello di trasporto (demux TCP o UDP)

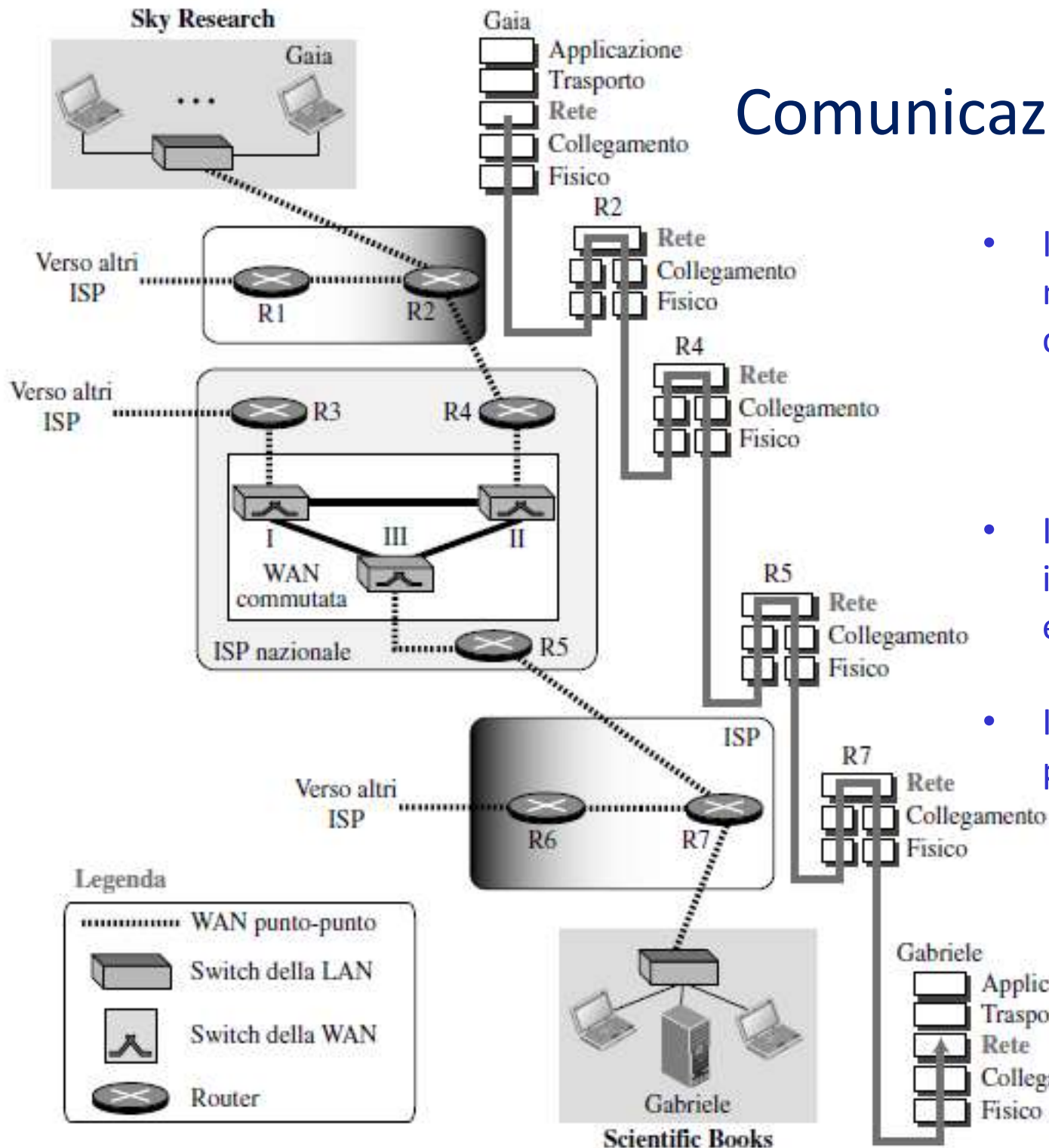
# Comunicazione a livello rete



Quali differenze notate rispetto al livello di trasporto?

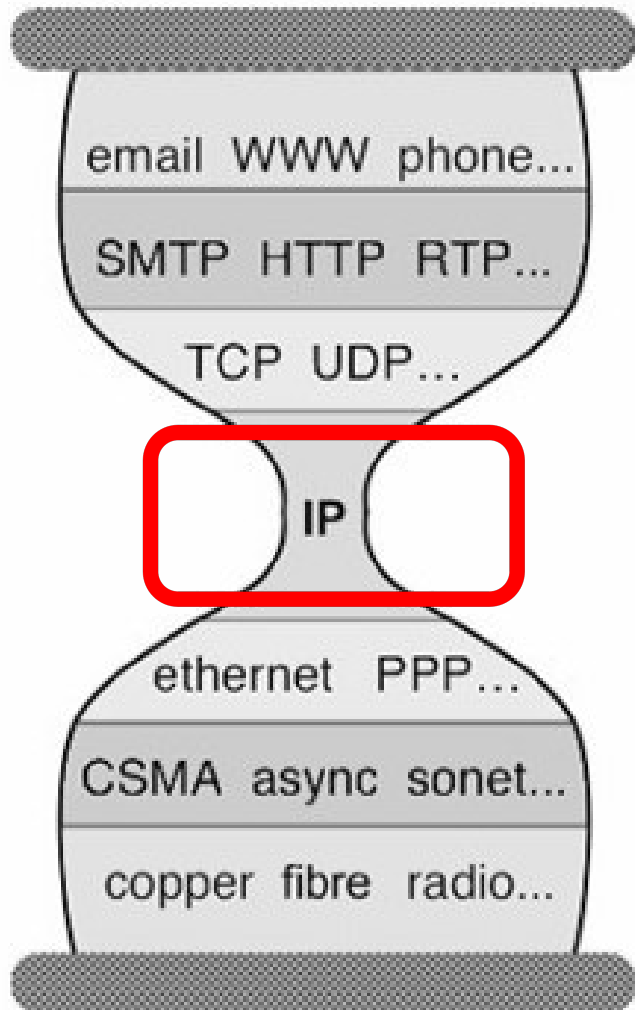
Quali sono le funzioni essenziali del livello di rete?

# Comunicazione a livello rete



- Il livello di rete è presente negli host mittente e destinatario e nei router
  - (livello trasporto solo negli host terminali)
- Il livello di rete interconnette reti eterogenee
- Il livello rete implementa poche(!) funzioni

# Strato di Rete



## Interconnessione di reti eterogenee

- Internet: rete logica costituita da un insieme di reti fisiche.
- Il livello di rete offre un'astrazione che consente a host e reti eterogenee di funzionare dal punto di vista logico come una singola rete

## Modello a clessidra

- Innovazione possibile nei livelli sottostanti
- Innovazione possibili ai livelli superiori (in particolare applicazioni)

Semplice è bello ma...

nel tempo il modello si è trasformato...

# Servizi a livello di rete

- Suddivisione in pacchetti
- Instradamento (routing)
- Inoltro (forwarding)

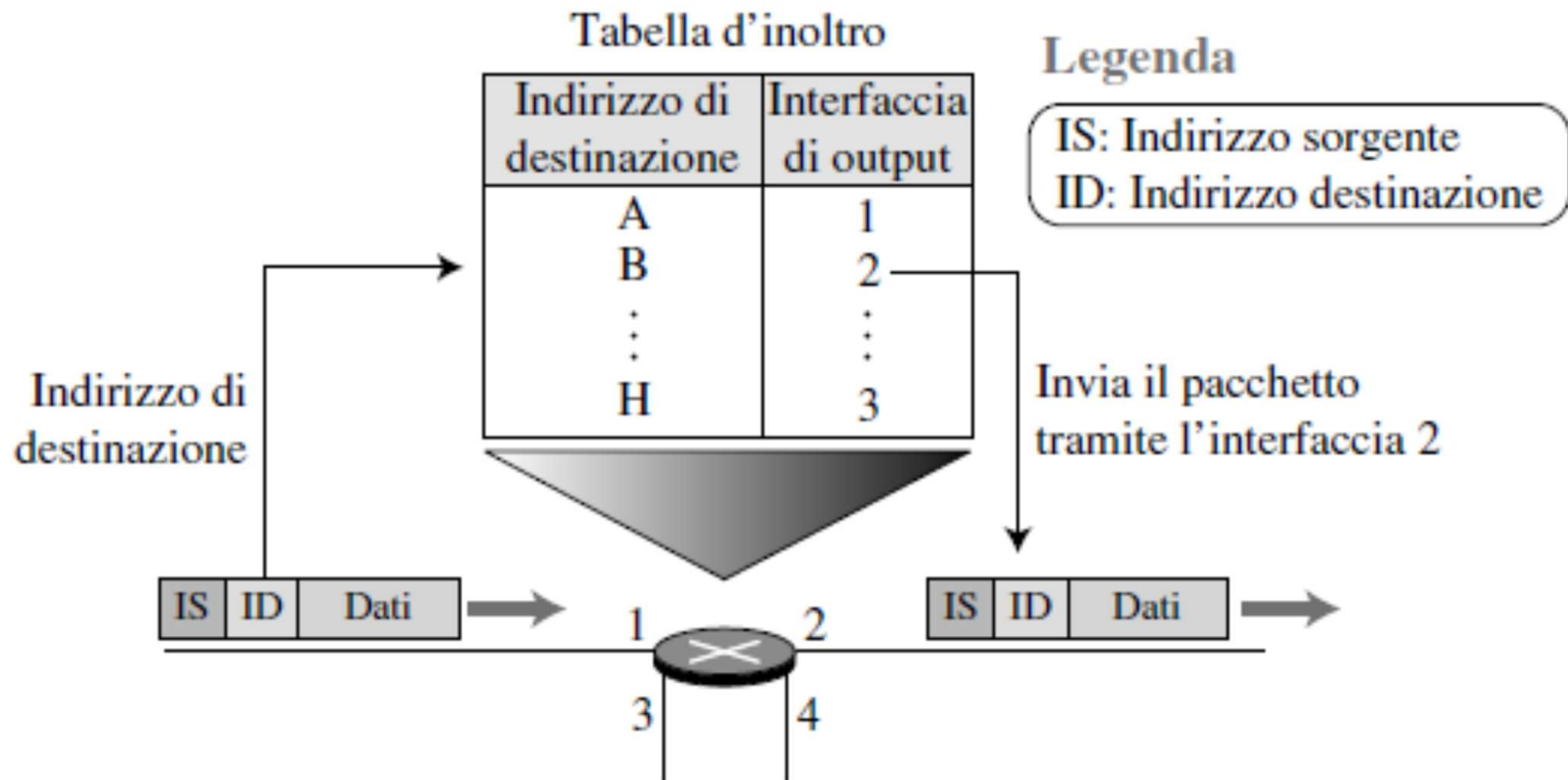
# Internet Protocol (IP) RFC 791

- Protocollo di tipo connection-less, non c'è circuito virtuale né fisico fra i due sistemi terminali a livello IP
- Servizio best effort (massimo impegno possibile):
  - non c'è garanzia che i pacchetti vengano ricevuti nell'ordine in cui sono stati inviati
  - non è garantita la consegna
- E' non affidabile, cioè non prevede meccanismi di recupero di errore: "Send and Pray"!!!
- Non prevede garanzie sulla Qualità del Servizio (QoS), sul tempo di consegna dei datagrammi e sul controllo di flusso
- Servizi aggiuntivi mediante estensioni
  - Modelli di QoS implementati dai router (es. DiffServ)

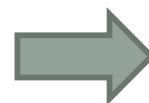


# Funzioni IP 1/2

- Inoltro (Forwarding): trasferimento del pacchetto sull'appropriato collegamento di uscita



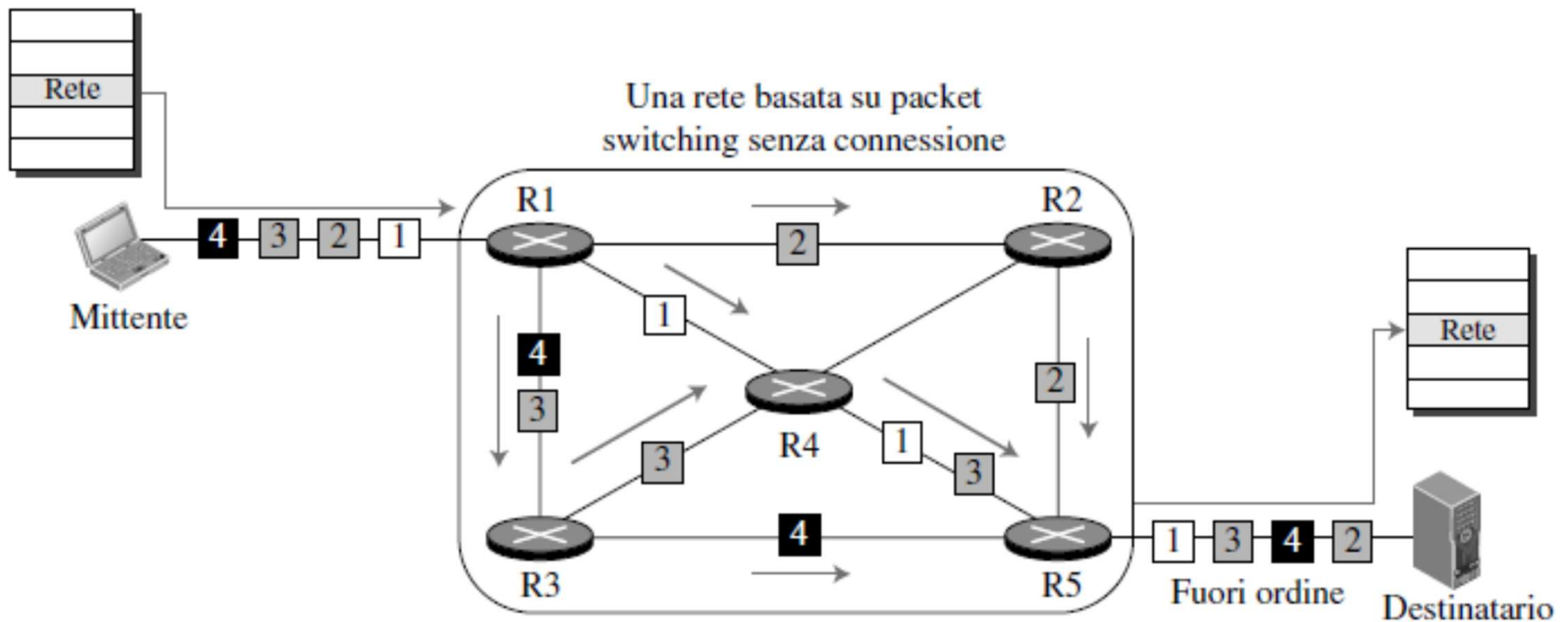
Problema: come costruire la  
tabella di inoltro?



Funzione di instradamento

# Funzioni IP 2/2

- Instradamento: processo decisionale di scelta del percorso verso una destinazione (algoritmi di instradamento o *routing*)

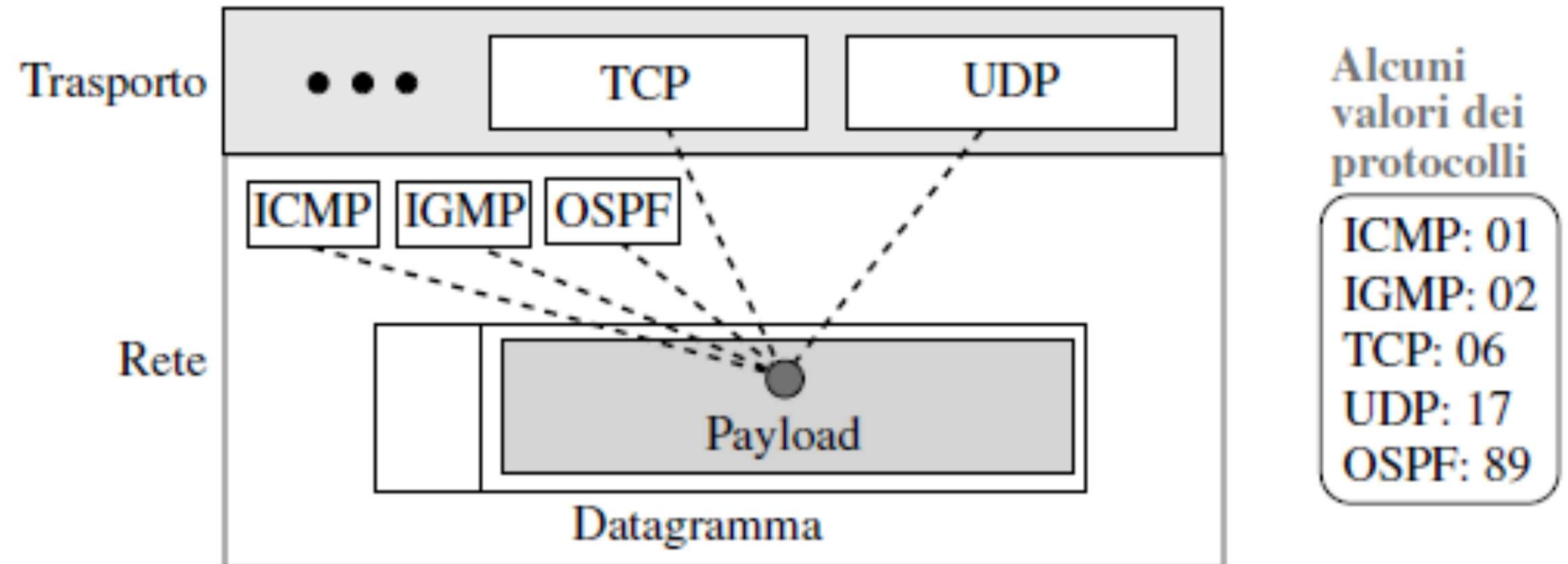


# Meccanismi IP

- Indirizzamento
  - Strumento per identificare gli host nell'internet
- Modello datagram (senza connessione) per la consegna dei dati
  - Frammentazione e riunificazione dei pacchetti (adattamento al data-link)
  - Controllo degli errori (solo header)
  - Verifica TTL

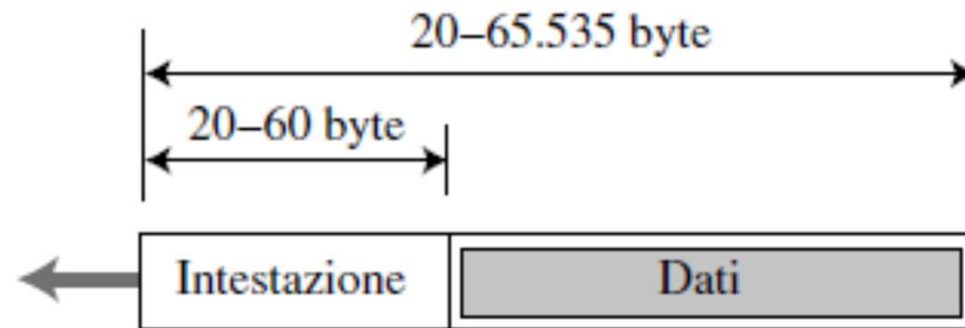
# Multiplexing/demultiplexing

NB anche a livello rete si fa  
multiplexing/demultiplexing

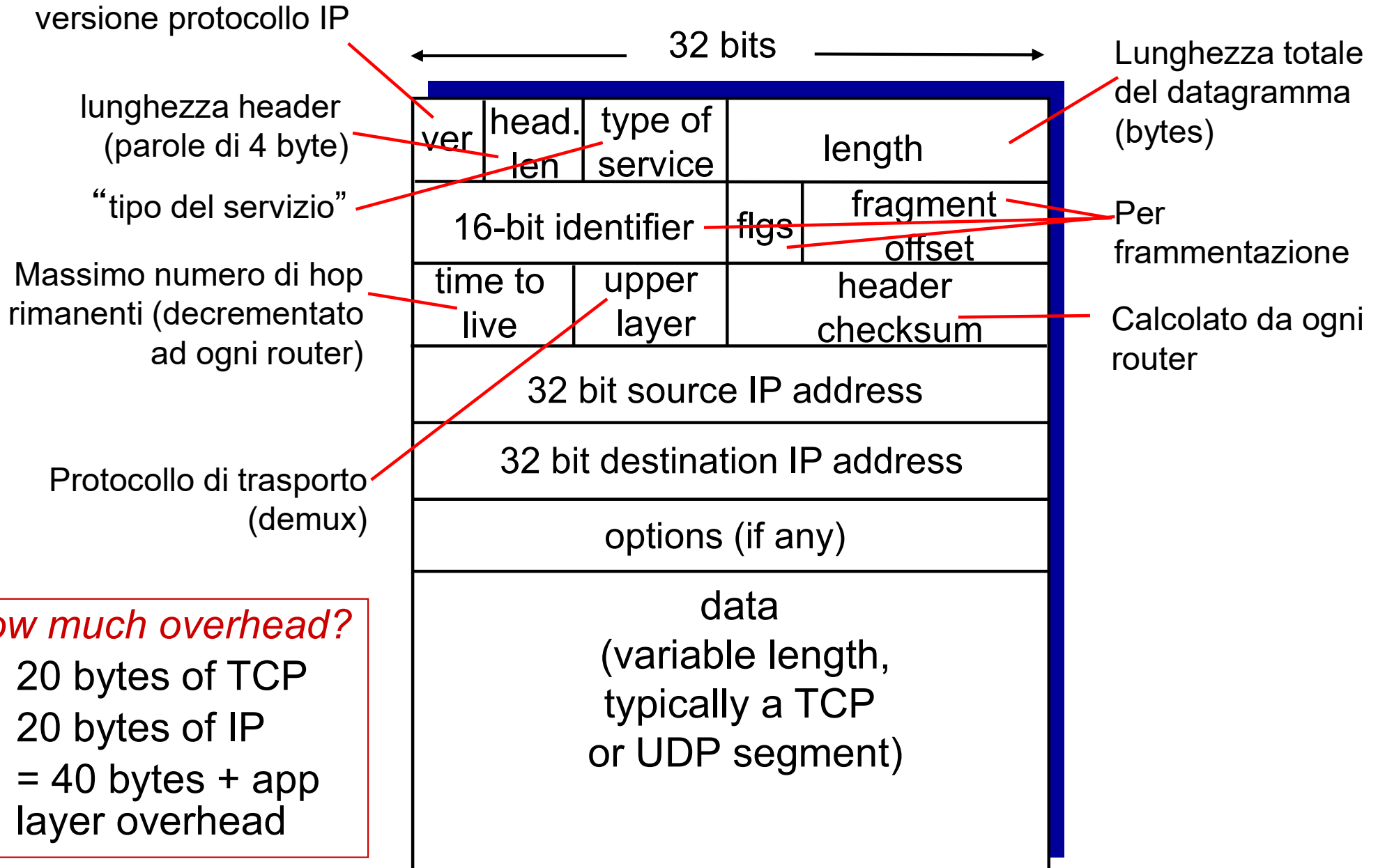


# Formato datagramma IP

# Datagramma IP



# IP datagram format



# Datagramma IP: campi

- **Versione (4 bit)**: specifica la versione di IP usata (attualmente IPv4 o IPv6)
- **Hlen (lunghezza dell'header, 4 bit)**: lunghezza dell'intestazione espressa in parole da 32 bit. Tipicamente vale 0101 ovvero 20 byte.
- **Tipo di servizio (8 bit)**: serve per “colorare” il datagramma IP (basso ritardo, affidabilità, ecc...). Inizialmente Type of Service, poi :
  - 6 bit per Differentiated Services
    - I pacchetti vengono marcati in base alla classe di servizio (telefonata, controllo, multimedia streaming, dati a bassa priorità)
    - Differenti politiche di accodamento ai router
  - 2 bit Explicit Congestion Notification (ECN) – supporto a livello rete e trasporto per la notifica di eventi di congestione



# Datagramma IP: campi

- **Lunghezza del datagramma (16 bit)**: è la lunghezza di tutto il datagramma in byte, header incluso.
  - Dim. Max 65535 byte (nella pratica 1500 byte)
- **Identificatore, flag, offset**: sono campi per la frammentazione (vedi più avanti).
- **Tempo di vita (8 bit)**: ad ogni passaggio da un router viene decrementato, quando raggiunge lo zero viene scartato. Assicura che eventuali percorsi ad anello non provochino traffico perpetuo nella rete
  - Valore iniziale dipende dal S.O.: esempi 30, 64, 128, 255

# Datagramma IP: campi

- **Protocollo (8 bit)**: in ricezione all'host destinatario indica quale protocollo dello strato superiore deve ricevere i dati. Inizialmente la tabella era nella RFC1700 (STD2), attualmente è un database online che si trova su [iana.org](http://iana.org)
  - Demux, es. TCP 6, UDP 17
- **Checksum dell'intestazione (16 bit)**: viene calcolato il checksum della sola intestazione (ponendo questo campo pari a zero) ad ogni router (il TTL cambia ad ogni hop!). Se si ottiene un errore si scarta il datagramma.

# Datagramma IP: campi

- **Indirizzi sorgente e destinazione (32 bit)**: indirizzi IP del mittente e del destinatario.
- **Opzioni (lunghezza variabile, multiplo di 32 bit)**: uso sporadico, da 0 a 40 byte
- **Dati**: dati trasportati dal datagramma IP.

### ***Approfondimento suggerito***

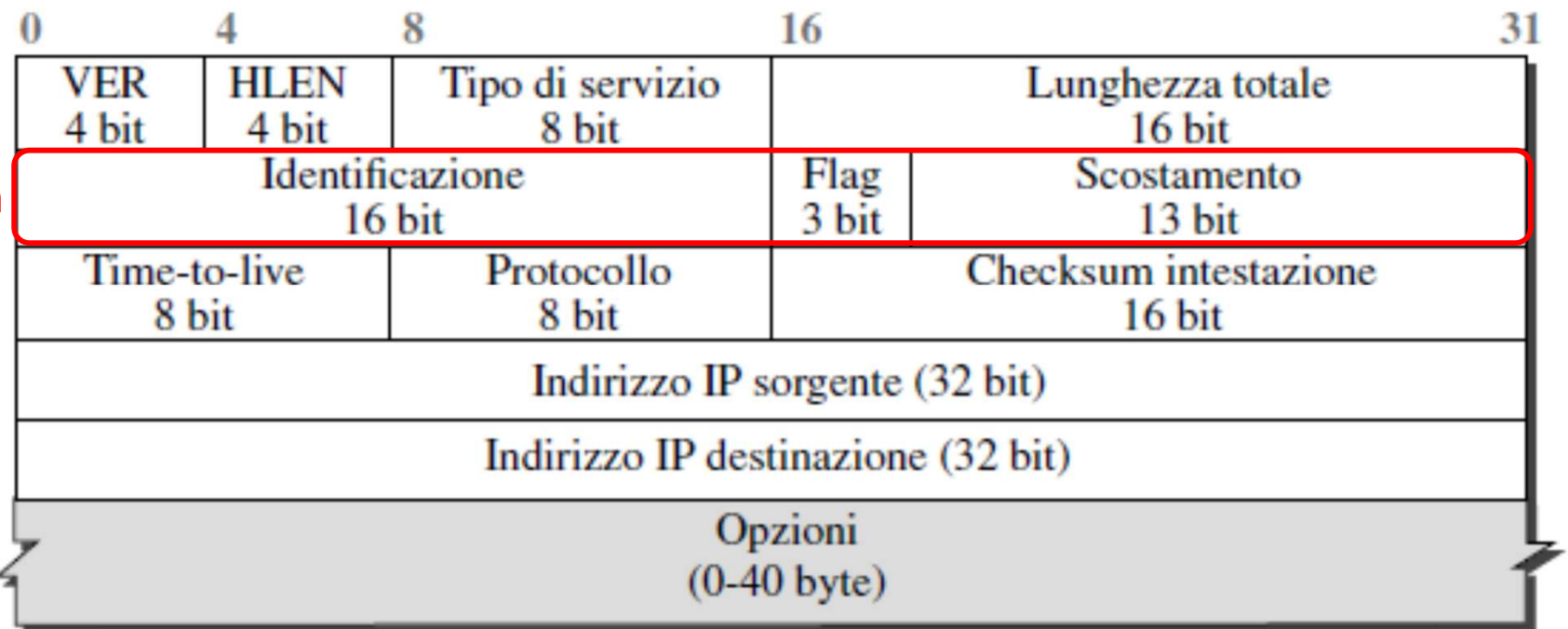
Determinare per quale motivo UDP e TCP si preoccupano di controllare l'integrità di informazioni di livello network se anche IP usa una sua checksum.

# Frammentazione

## Legenda

VER: numero di versione

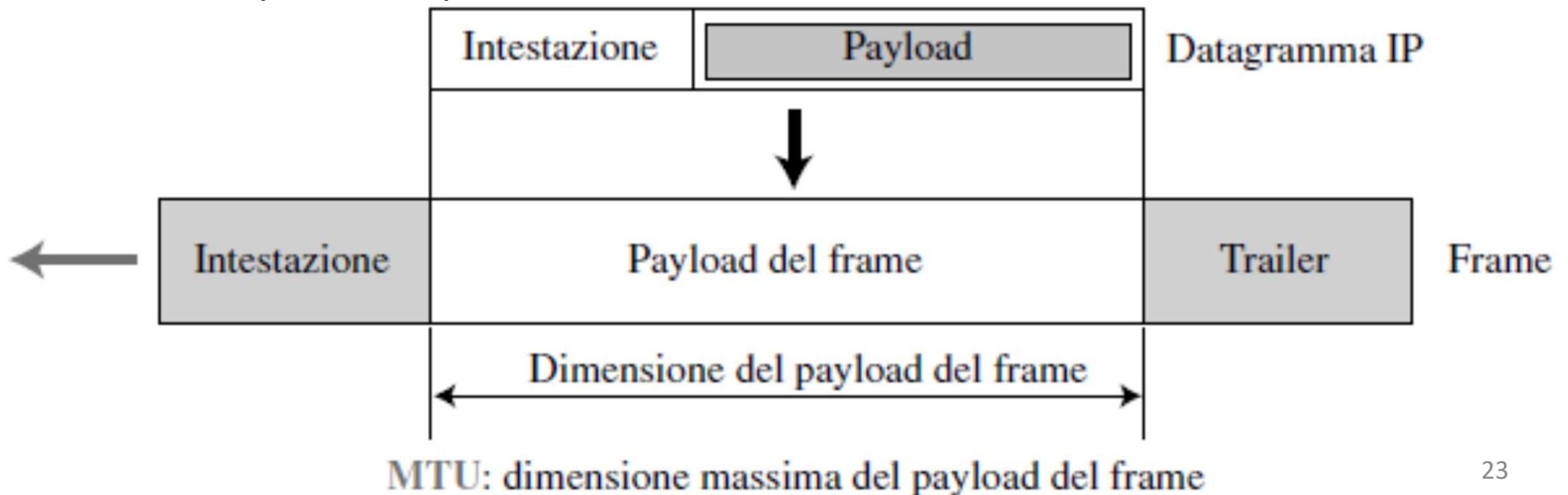
HLEN: lunghezza dell'intestazione  
byte: 8 bit



fragmentation

# Frammentazione

- MTU (Maximum Transfer Unit)
  - quantità massima di dati trasportata dal protocollo di collegamento (in un frame). Può variare da una tecnologia di collegamento all'altra (ethernet 1500, ATM 576, FDDI 4500 byte ecc.).
- La MTU pone un limite alla lunghezza dei datagrammi IP e tratte diversi possono porre limiti differenti.



# Frammentazione

- Meccanismo di frammentazione
  - Se il router riceve un datagramma la cui dimensione supera l'MTU della rete verso cui deve inoltrare quel datagramma, **il router frammenta il datagramma IP in due o più datagrammi più piccoli**, detti frammenti.
  - Il riassemblaggio viene effettuato dall'entità rete nel sistema terminale (destinazione).
  - **Se qualche frammento non arriva a destinazione?**
    - si butta via tutto il datagramma
- **Un pacchetto diviso in n frammenti. Come sono inoltrati questi frammenti?**
  - Ciascun frammento è a sua volta un datagramma IP completo che viene trasmesso attraverso una serie di reti fisiche indipendentemente dagli altri

# FRAMMENTAZIONE IP

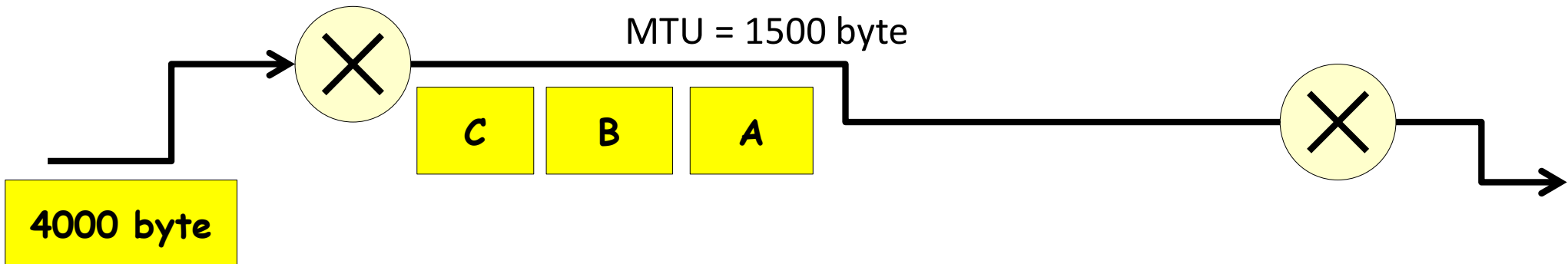
- I frammenti possono arrivare fuori ordine, come fa il livello di rete a ricomporre i frammenti nel pacchetto originale?
- Usa identificazione, flag, e offset
  - **Identificatore (16 bit)** è associato al datagramma dall'host sorgente. Associato a IP sorgente e IP destinazione identifica quel datagramma in un intervallo di tempo ragionevolmente lungo.
    - i frammenti di quel datagramma mantengono il valore di questo campo. Il destinatario riconosce i frammenti che vanno assemblati insieme
  - **Offset (13 bit)** indica la posizione relativa come multiplo di 8 byte. Serve ad ordinare i frammenti nell'assemblaggio.
    - I frammenti devono essere multipli di 8 byte
    - Primo frammento Offset =0
  - **Flag (3 bit)** serve ad identificare l'ultimo frammento – vedi slide seguente



# FRAMMENTAZIONE IP – campo flag

- Flag
  - Il bit 0 è riservato (per ora vale sempre 0)
  - Il bit 1 *do not fragment* vale:
    - 0 se il pacchetto può essere frammentato
    - 1 se il pacchetto non deve essere frammentato
  - Il bit 2 *more fragments* vale:
    - 0 se il pacchetto è l'ultimo frammento
    - 1 se il pacchetto non è l'ultimo frammento

# FRAMMENTAZIONE IP –esempio



Arriva un datagramma di 4000 byte (header IP incluso).  
Considerando 20 byte di intestazione IP

quanti byte nel campo dati?  
1500 con header di 20 byte  
1480 byte  
Calcolo offset  
 $1480/8=185$   
1480 è multiplo di 8!

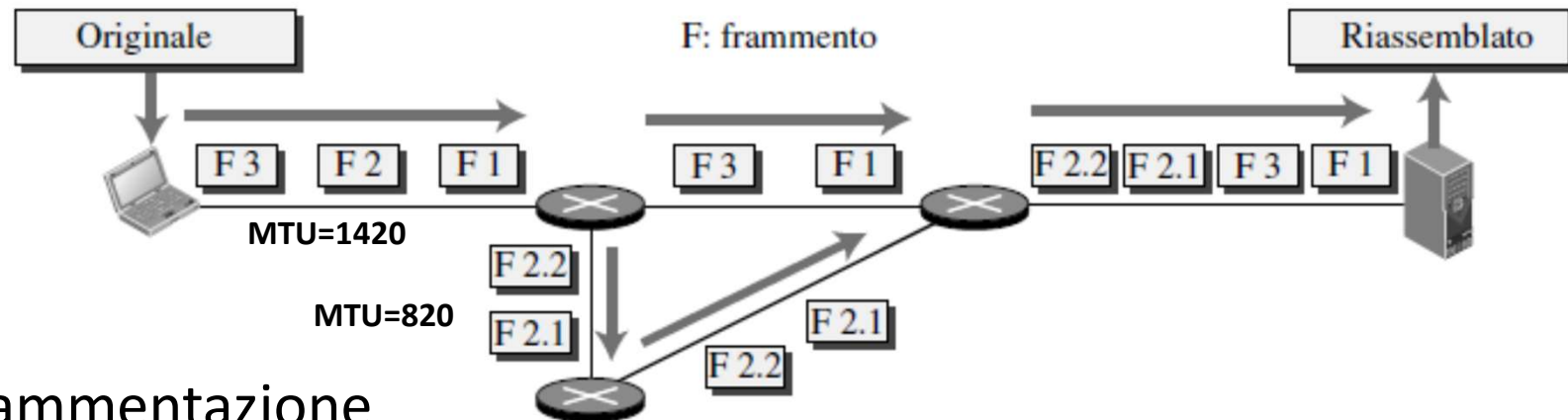
offset =

	length	ID	fragflag	offset	
	=4000	=x	=0	=0	

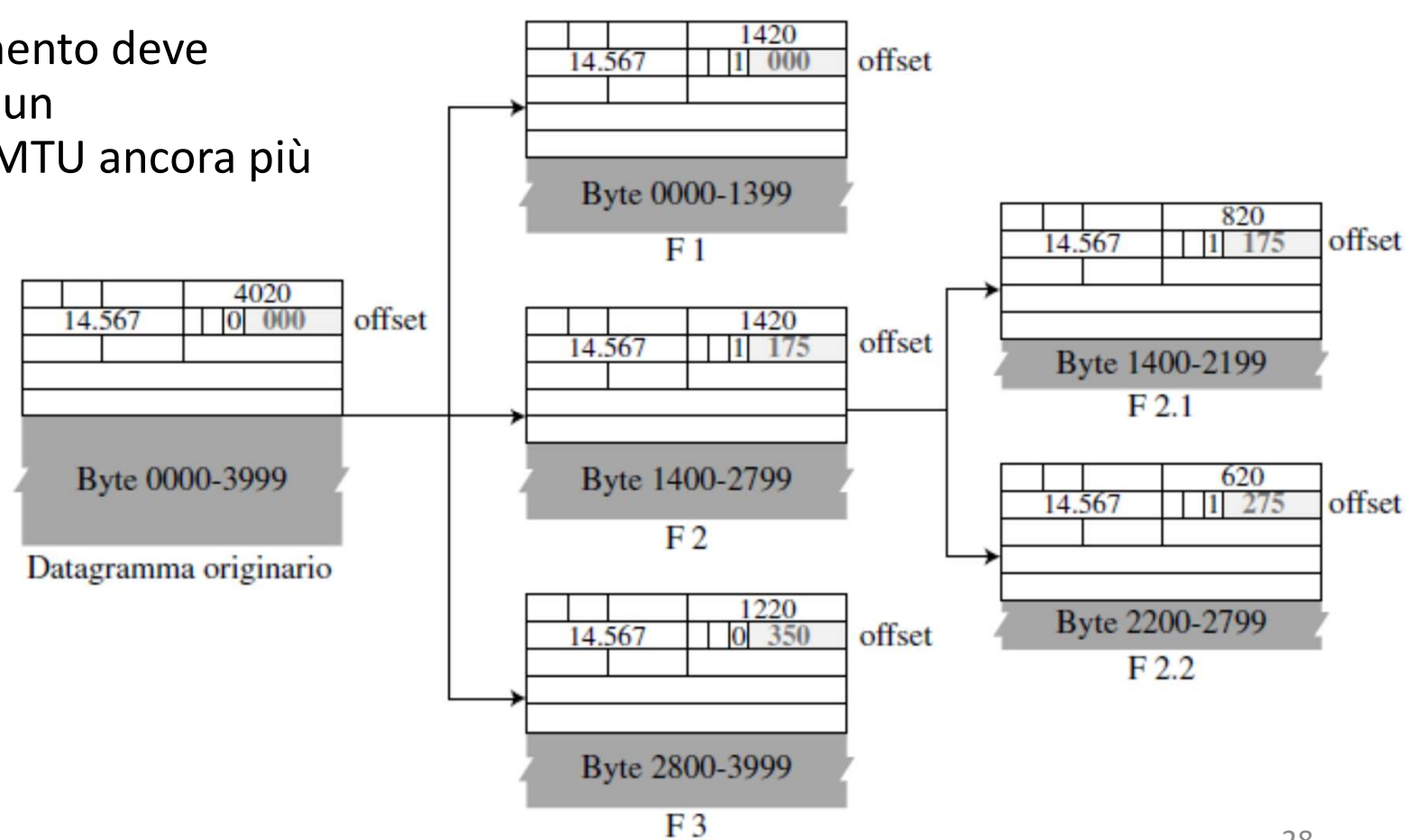
	length	ID	fragflag	offset	0-1479
	= 1500	= x	= 1	= 0	

	length	ID	fragflag	offset	1480 – 2959
	= 1500	=x	= 1	= 185	

	length	ID	fragflag	offset	2960- 3980
	= 1040	=x	= 0	= 370	



- Il processo di frammentazione può essere ripetuto
- Ad es. se un frammento deve essere inoltrato su un collegamento con MTU ancora più piccolo



# Frammentazione IP

- Ricostruzione del datagramma è un processo critico
  - Perché?
- Soluzioni?

# Frammentazione IP

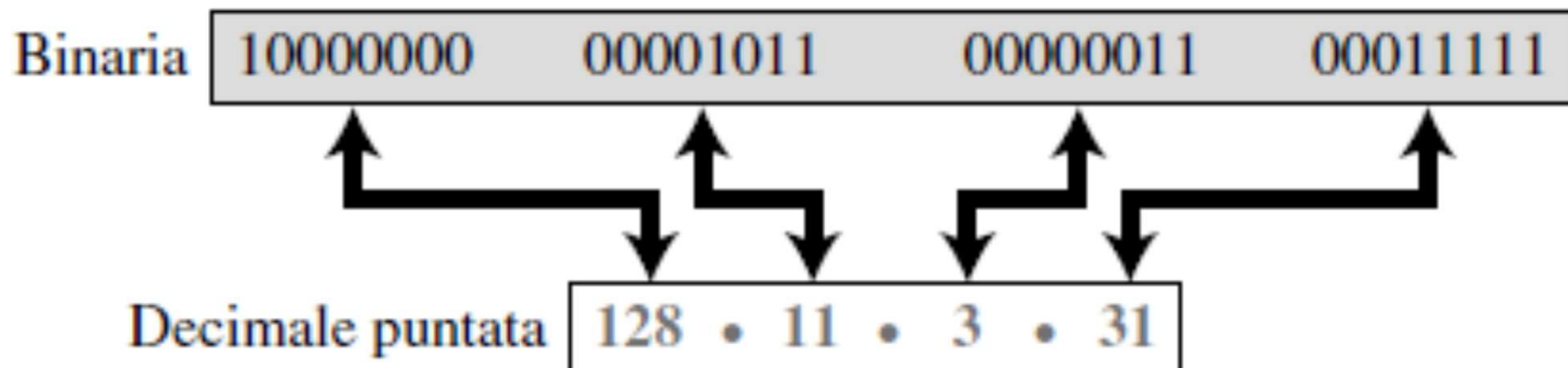
- Ricostruzione del datagramma è un processo critico
  - richiede risorse ai nodi intermedi e destinazione
  - Ritardo di elaborazione
  - se un frammento manca il datagramma deve essere scartato.
- Soluzioni?
  - Non frammentare!
  - Do not fragment in IPv4
  - IPv6 non supporta la frammentazione



# INDIRIZZAMENTO IP

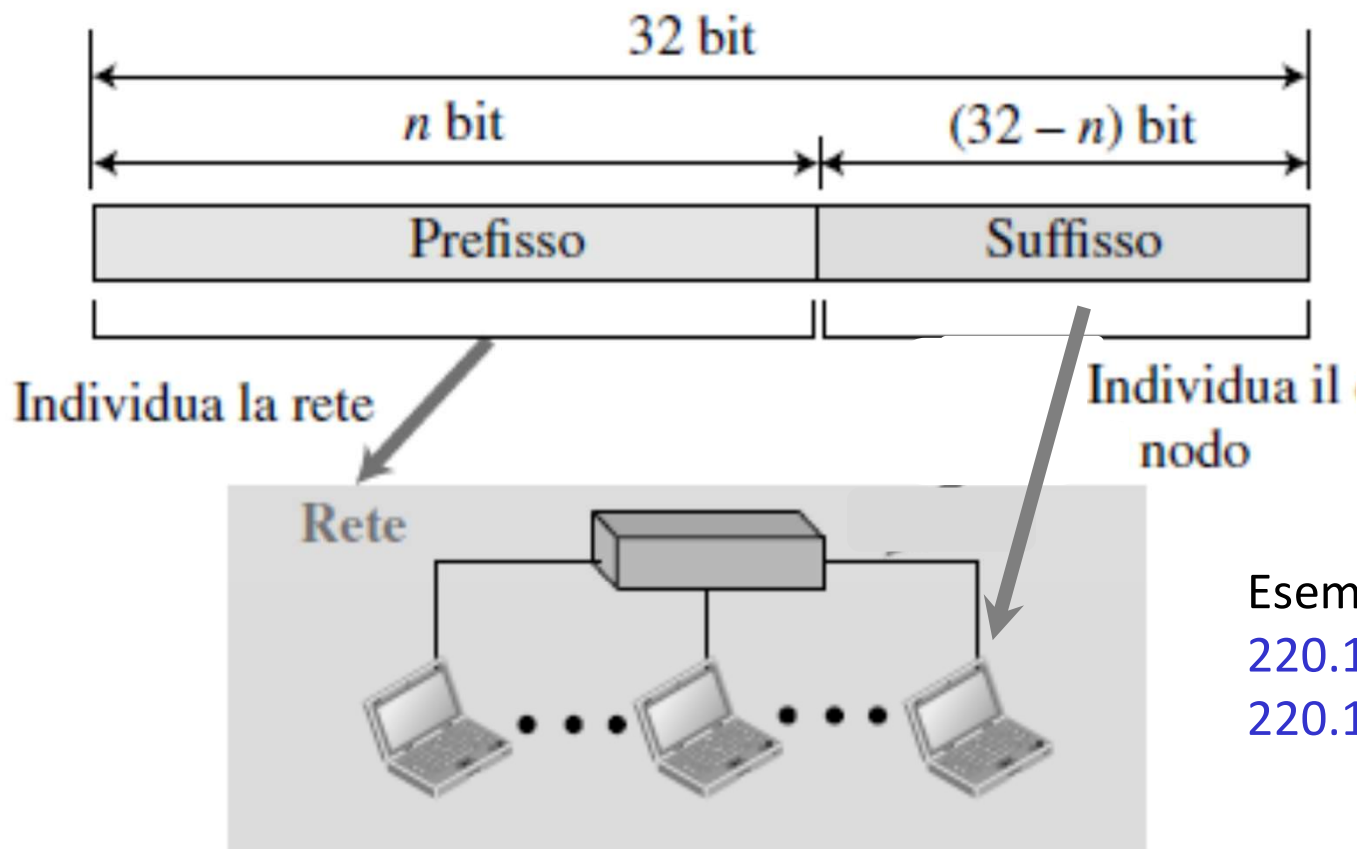
# Indirizzi IPv4

- Ogni host è connesso ad Internet attraverso un'interfaccia di rete, che è il confine fra l'host ed il collegamento su cui vengono inviati i datagrammi.
  - Ad ogni interfaccia è assegnato un indirizzo IP.
  - I router devono necessariamente essere connessi ad almeno due collegamenti.
- Gli indirizzi IP sono costituiti da 32 bit (4 byte), rappresentati in notazione decimale puntata.



# Indirizzi IP

- Ogni host ha un indirizzo univoco diviso in due parti:
- **Network ID + Host ID** , che identificano una rete IP su Internet e l'host su quella rete IP.





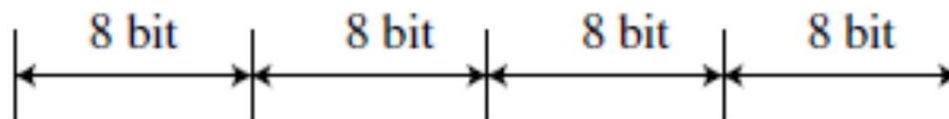
# Strategie di indirizzamento

- Come assegnare gli indirizzi?
- Come identificare in un indirizzo IP gli *n bit* che identificano la rete?

# Classful addressing

- Prefissi di rete di lunghezza fissa
- Reti divise in classi

Spazio degli indirizzi: 4.294.967.296 indirizzi



Classe A	0	Prefisso	Suffisso
Classe B	10	Prefisso	Suffisso
Classe C	110	Prefisso	Suffisso
Classe D	1110	Indirizzi multicast	
Classe E	1111	Riservati per uso futuro	

Classe	Prefissi	Primo byte
A	$n = 8$ bit	Da 0 a 127
B	$n = 16$ bit	Da 128 a 191
C	$n = 24$ bit	Da 192 a 223
D	Non applicabile	Da 224 a 239
E	Non applicabile	Da 240 a 255

# Classful addressing

- 5 classi di indirizzi IP
  - Classe A: 7 bit per 128 reti IP, 24 bit
    - 0.0.0.0 - 127.255.255.255
    - Reti di 16 milioni di host
  - Classe B: 14 bit per ca. 16000 reti IP, 16 bit host id
    - 128.0.0.0 - 191.255.255.255
    - Reti di circa 64000 host
  - Classe C: 21 bit per reti IP e 8 bit per host id
    - 192.0.0.0 - 223.255.255.255
    - Reti di circa 256 host
  - Classe D, riservata a multicasting,
    - 224.0.0.0 - 239.255.255.255
  - Classe E, riservata per usi futuri/ricerca
    - 240.0.0.0 – 255.255.255

# Esempi di Indirizzi IP

- **15.10.10.90**: E' un indirizzo di classe A, poiché il primo numero è compreso fra 0 e 127, dunque i seguenti campi indicano l'host 10.10.90 nella rete IP 15.
- **130.250.42.53**: E' un indirizzo di classe B, poiché il primo numero è compreso fra 128 e 191; dunque i primi due campi indicano la rete IP 130.250, gli altri indicano l'host 42.53.
- **196.234.12.14**: E' un indirizzo di classe C, poiché il primo numero è compreso fra 192 e 223; dunque i primi tre campi indicano la rete IP 196.234.12, e l'ultimo indica l'host 14.

# Strategie di addressing

- Classful addressing
  - Dato un indirizzo IP è facile risalire all'indirizzo di rete
  - Ma... poco flessibile nell'utilizzo dello scarso range di indirizzi disponibili
- Soluzioni?

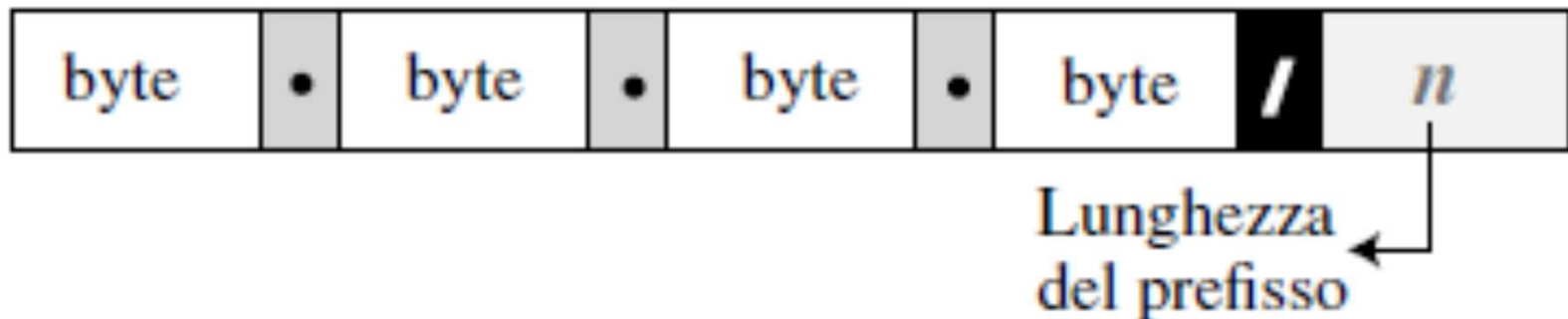
# Strategie di addressing

- Classful addressing
  - Dato un indirizzo IP è facile risalire all'indirizzo di rete
  - Ma... poco flessibile nell'utilizzo dello scarso range di indirizzi disponibili
- Soluzioni?
  - Classless addressing
  - IPv6: indirizzi IP di 128 bit anziché 32 bit
    - Prefix length (multipli di 4 bit)



# Indirizzamento senza classi

- Notazione CIDR (Classless Interdomain Routing) RFC 1519



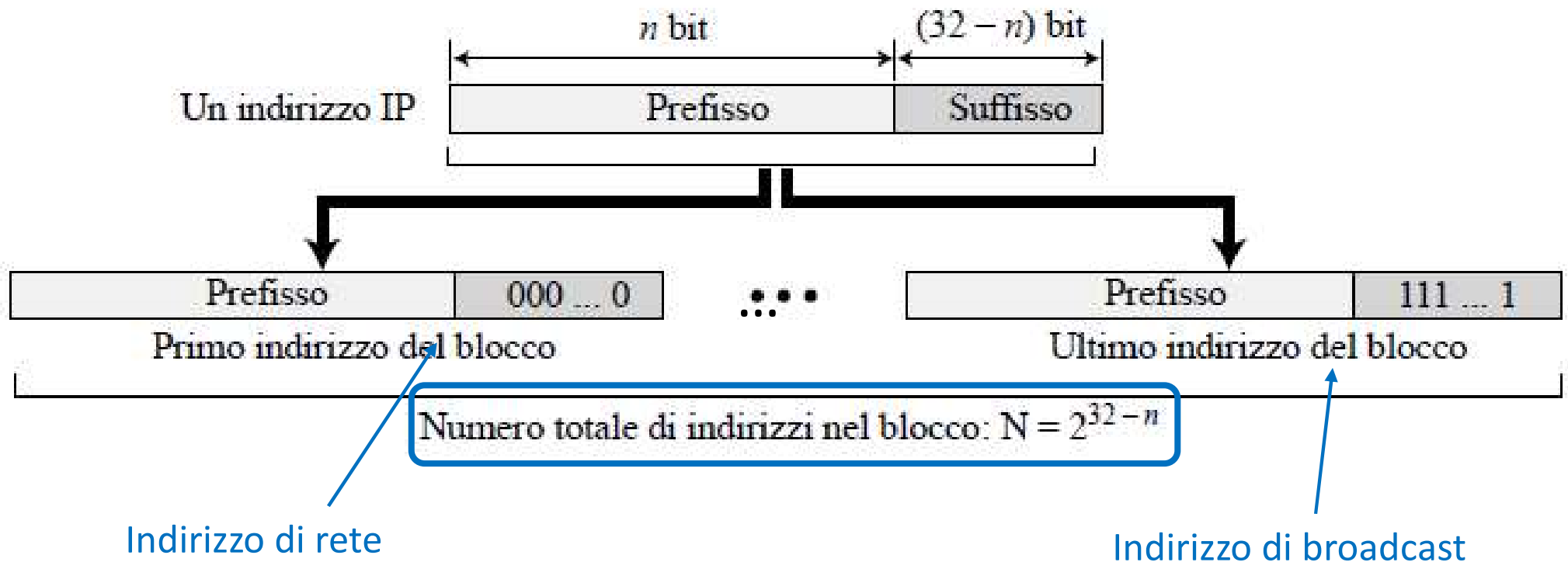
Esempi:

12.24.76.8/8

23.14.67.92/12

220.8.24.255/25

# Indirizzamento senza classi



- Numero massimo di host?
  - Una rete con un prefisso di rete di  $n$  bit può avere al massimo  $2^{32-n} - 2$  host (devo sottrarre indirizzo di rete e indirizzo di broadcast)

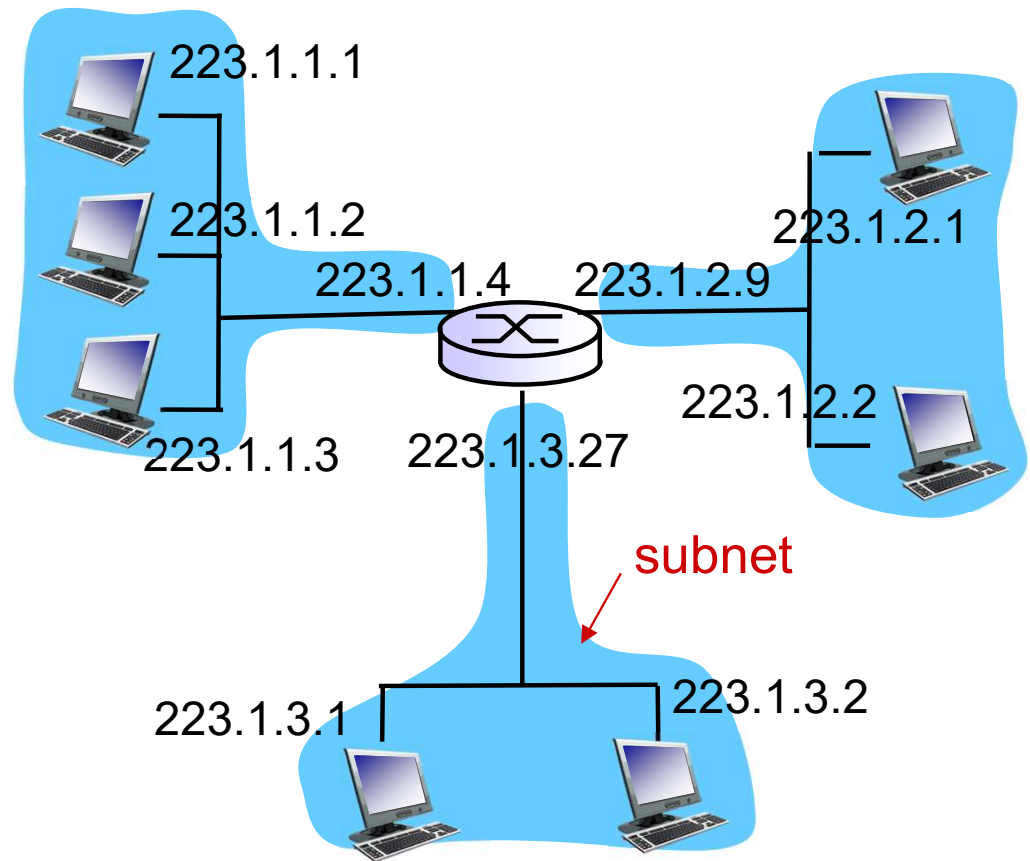


# Esempio

Rete data  
dall'interconnessione di tre  
sottoreti

24 bit per l'indirizzo di rete:  
/24

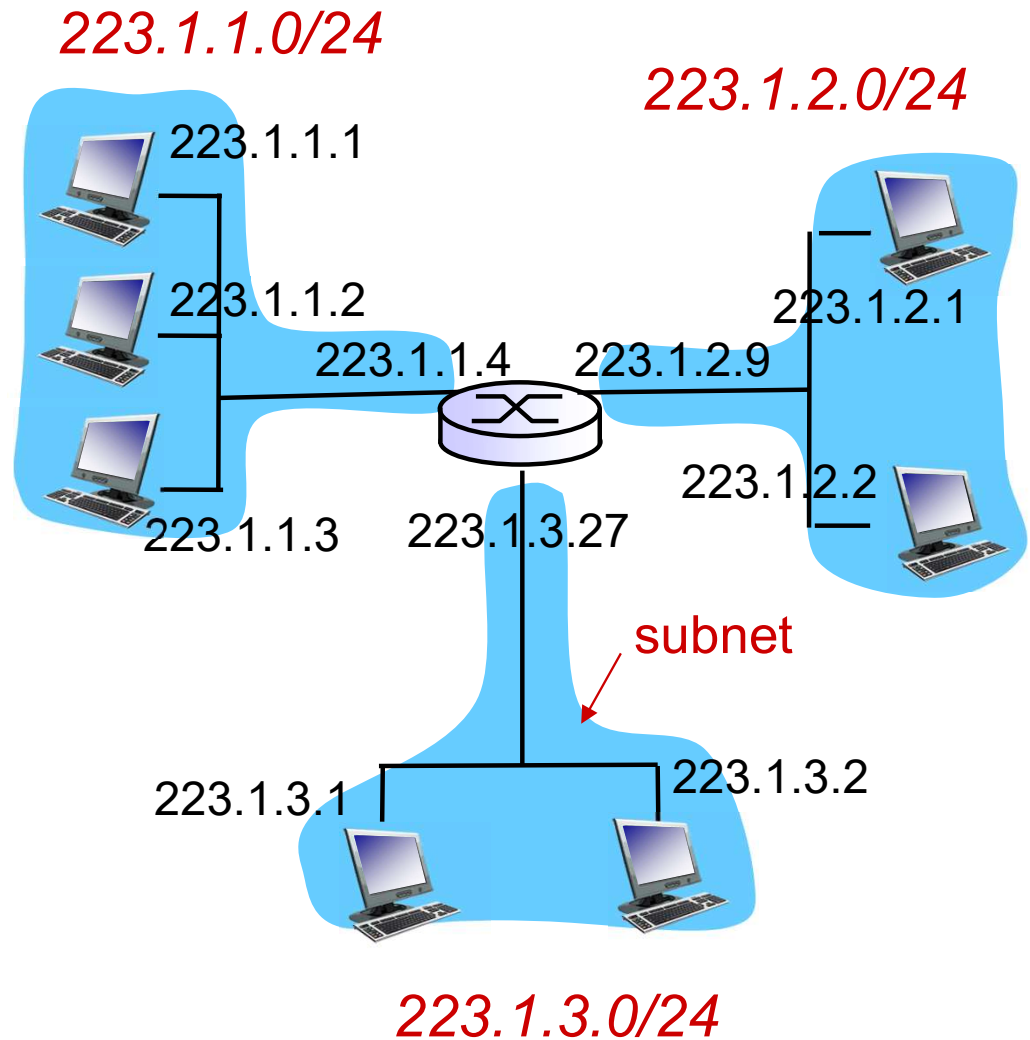
Indicare l'indirizzo di  
ciascuna sottorete



# Esempio

Rete data  
dall'interconnessione di tre  
sottoreti

24 bit per l'indirizzo di rete:  
/24



# Maschera di sottorete

Indirizzo

$a.b.c.d/n$

$n$  bit più a sinistra costituiscono la parte di indirizzo di rete

Subnet mask (maschera di sottorete)

Numero composto da 32 bit in cui i primi  $n$  bit a sinistra sono impostati a 1 e i rimanenti a 0

Serve a distinguere quale parte di un indirizzo ip identifica la rete e quale l'host

# Subnet mask e indirizzo di rete: esempio

- Ipotesi: l'indirizzo IP 150.217.8.42 ha netmask 255.255.255.0
- Qual è l'indirizzo di rete?

Si effettua un'operazione di AND

Indirizzo IP	150.217.8.42	10010110	11011001	00001000	00101010
Subnet Mask	255.255.255.0	11111111	11111111	11111111	00000000
AND	150.217.8.0	10010110	11011001	00001000	00000000

La rete ha indirizzo  
150.217.8.0/24

# Esercizio 1

Da quanto detto gli indirizzi:

150.193.4.1/22

150.193.8.18/22

150.193.11.3/22

appartengono alla stessa rete?

# Per rispondere, l'uomo.....

- Per verificare se i tre indirizzi IP appartengono alla stessa rete l'uomo confronta **i 22 bit più a sinistra** delle tre diverse sequenze:

150.193.4.1/22

**10010110 11000001 000001**00 00000001

150.193.8.18/22

**10010110 11000001 000010**00 00010010

150.193.11.3/22

**10010110 11000001 000010**11 00000011

e verifica così che il primo indirizzo non è sulla stessa rete degli altri due



# Per rispondere, il calcolatore. .

Indirizzo IP	150.193.4.1	10010110 11000001 00000100 00000001
subnet mask	255.255.252.0	11111111 11111111 11111100 00000000
Risultato AND	150.193.4.0	10010110 11000001 00000100 00000000

Indirizzo IP	150.193.8.18	10010110 11000001 00001000 00010010
subnet mask	255.255.252.0	11111111 11111111 11111100 00000000
Risultato AND	150.193.8.0	10010110 11000001 00001000 00000000

Indirizzo IP	150.193.11.3	10010110 11000001 00001011 00000011
subnet mask	255.255.252.0	11111111 11111111 11111100 00000000
Risultato AND	150.193.8.0	10010110 11000001 00001000 00000000



## Esercizio 2

Un host A ha l'indirizzo IP 167.199.170.82/27.

Indicare:

- l'indirizzo di rete di A
- il primo e ultimo indirizzo nel range degli indirizzi della rete di A