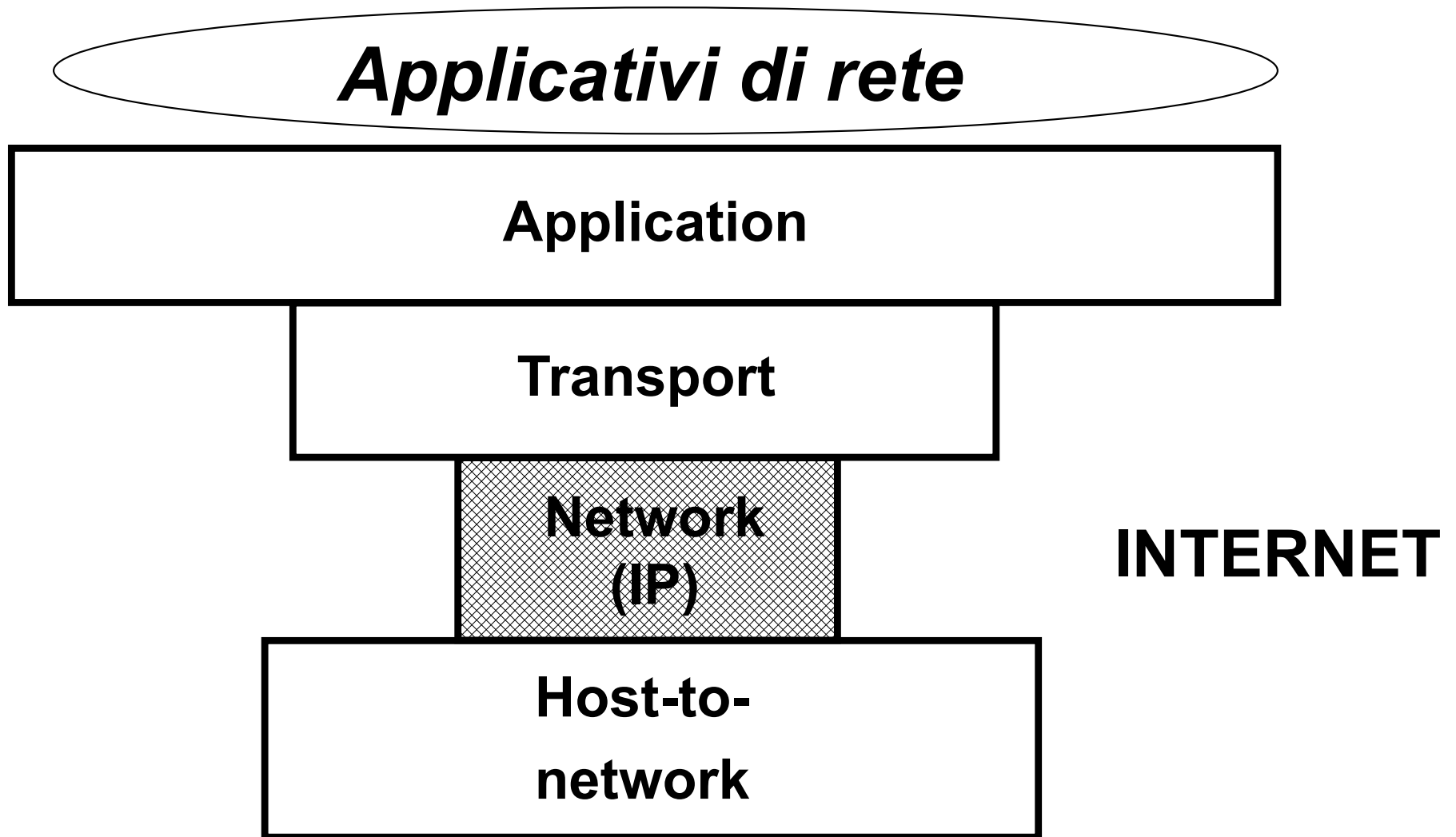


LIVELLO IP

(La “dorsale” di Internet)

Importanza del livello IP

Suite di protocolli TCP/IP



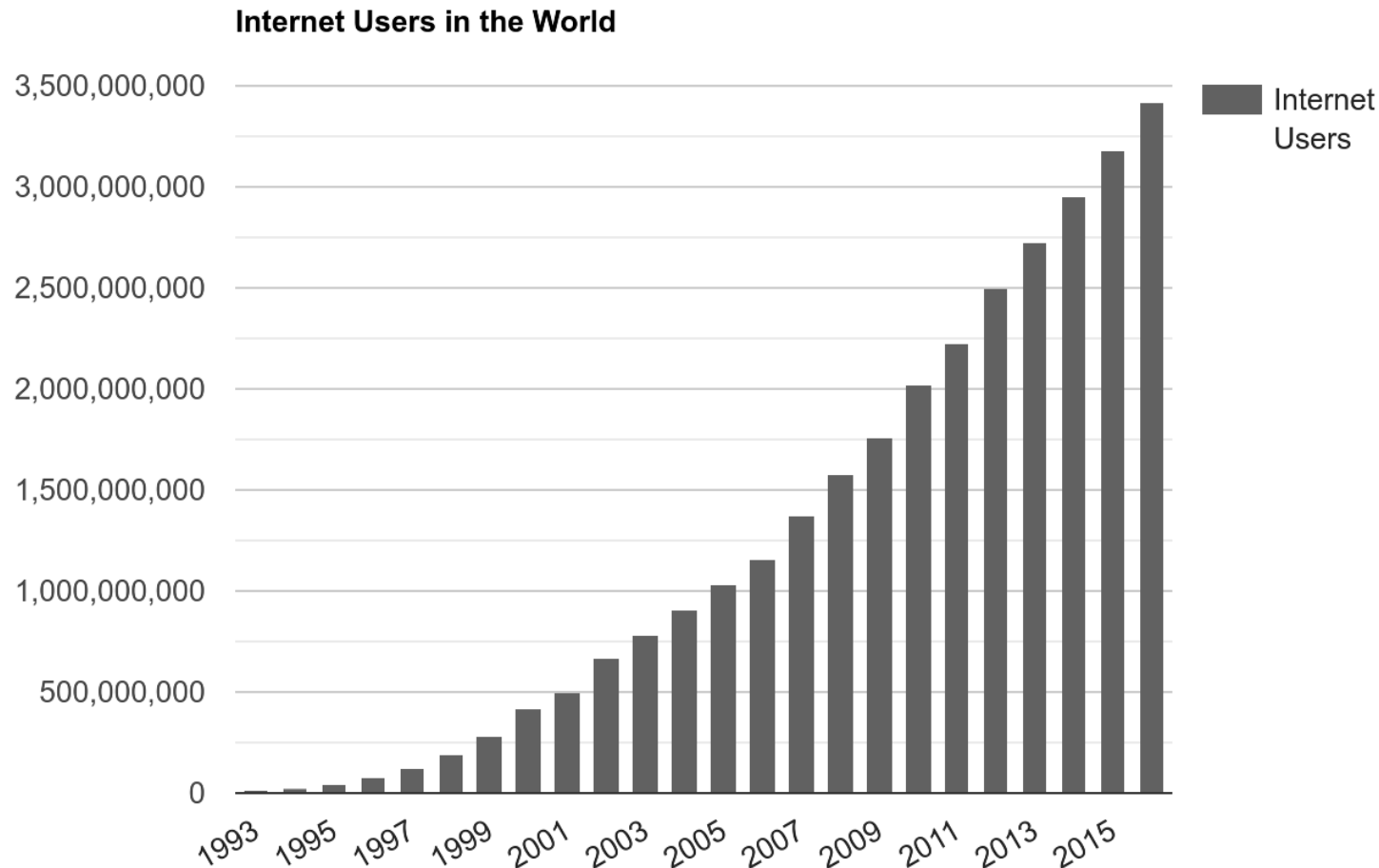
Il successo continuo e costante di Internet

**Tutti gli host
collegati ad
Internet devono
essere
“identificati” in
modo univoco**

1969	4
1979	200
1989	100.000
Gennaio 1993	1.313.000
Gennaio 1994	2.217.000
Gennaio 1995	4.852.000
Gennaio 1996	9.472.000
Gennaio 1997	16.146.000
Gennaio 1998	29.670.000
Gennaio 1999	43.230.000
Gennaio 2000	72.340.000
Gennaio 2001	109.574.000
Gennaio 2002	147.344.000
Gennaio 2003	171.638.000
Gennaio 2004	233.101.000
Gennaio 2005	317.646.000
Gennaio 2006	394.992.000
Gennaio 2007	433.194.000
Gennaio 2008	541.677.000
Gennaio 2009	625.226.000
Gennaio 2010	732.740.000
Gennaio 2011	818.374.000
Gennaio 2012	888.239.000
Gennaio 2013	963.518.000
Gennaio 2014	>miliardo!

Numero di host collegati ad Internet

La crescita esponenziale del numero di host in Internet



Numero di utenti connessi: > 5 miliardi

<https://www.internetlivestats.com/internet-users/>

<https://ourworldindata.org/internet>

IP: un protocollo “antico”

Descritto nell’RFC 791

Pubblicato dalla IETF nel settembre 1981

Ma cos'è Internet?

“Internet refers to the global information system that

- i) is logically linked together by a globally unique address space based on the Internet Protocol (IP) or its subsequent extensions/follow-ons;*
- ii) is able to support communications using the Transmission Control Protocol/Internet Protocol (TCP/IP) suite or its subsequent extensions/follow-ons, and/or other IP-compatible protocols;*
- iii) provides, uses or makes accessible, either publicly or privately, high level services layered on the communications and related infrastructure described herein”*

(Federal Networking Council)

Ma cos'è INTERNET ?

Una rete globale...

Obiettivo globale:

- Connettere un qualsiasi numero di reti (**scalabilità**) **eterogenee** (distinzione netta fra H2N e Network) e **indipendenti** (alta decentralizzazione organizzativa)

• Scelte fondamentali del progetto:

- Comunicazioni con paradigma ***packet switching***
- Nodi intermedi (**router**) che inoltrano i pacchetti con “intelligenza minima” (minore complessità = minori costi e maggiori performance)
 - Logica di inoltro **stateless**
 - Intelligenza delegata agli host (ai layer trasporto e applicativo, se necessaria)

Ma cos'è *INTERNET* ?

Un insieme di nodi e reti dotati di indirizzi univoci

Nodi terminali

- Host

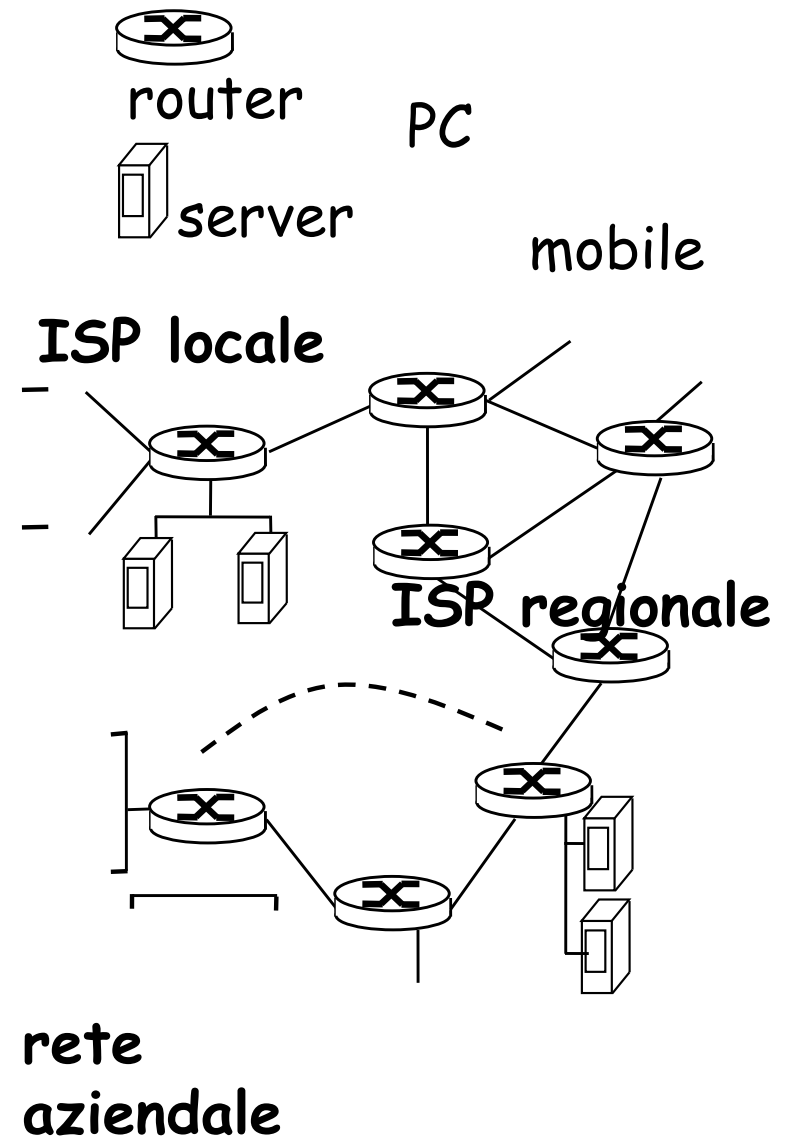
Nodi intermedi

- **Router**

Ogni nodo ha almeno un indirizzo IP univoco

I nodi sono “aggregati” in reti, identificabili a loro volta da “**blocchi di indirizzi**”

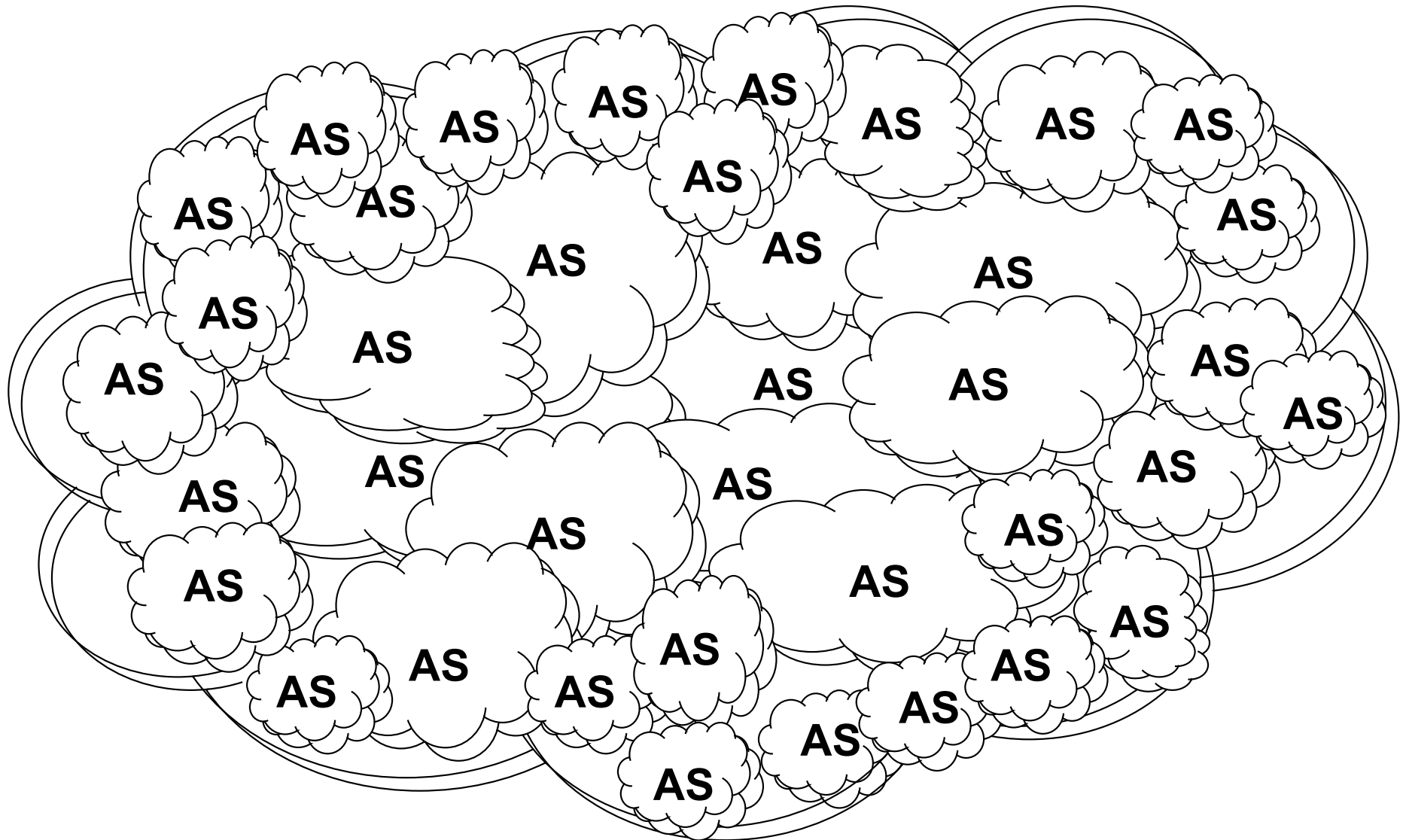
- Logica di **indirizzamento gerarchico**



Cos'è INTERNET ?

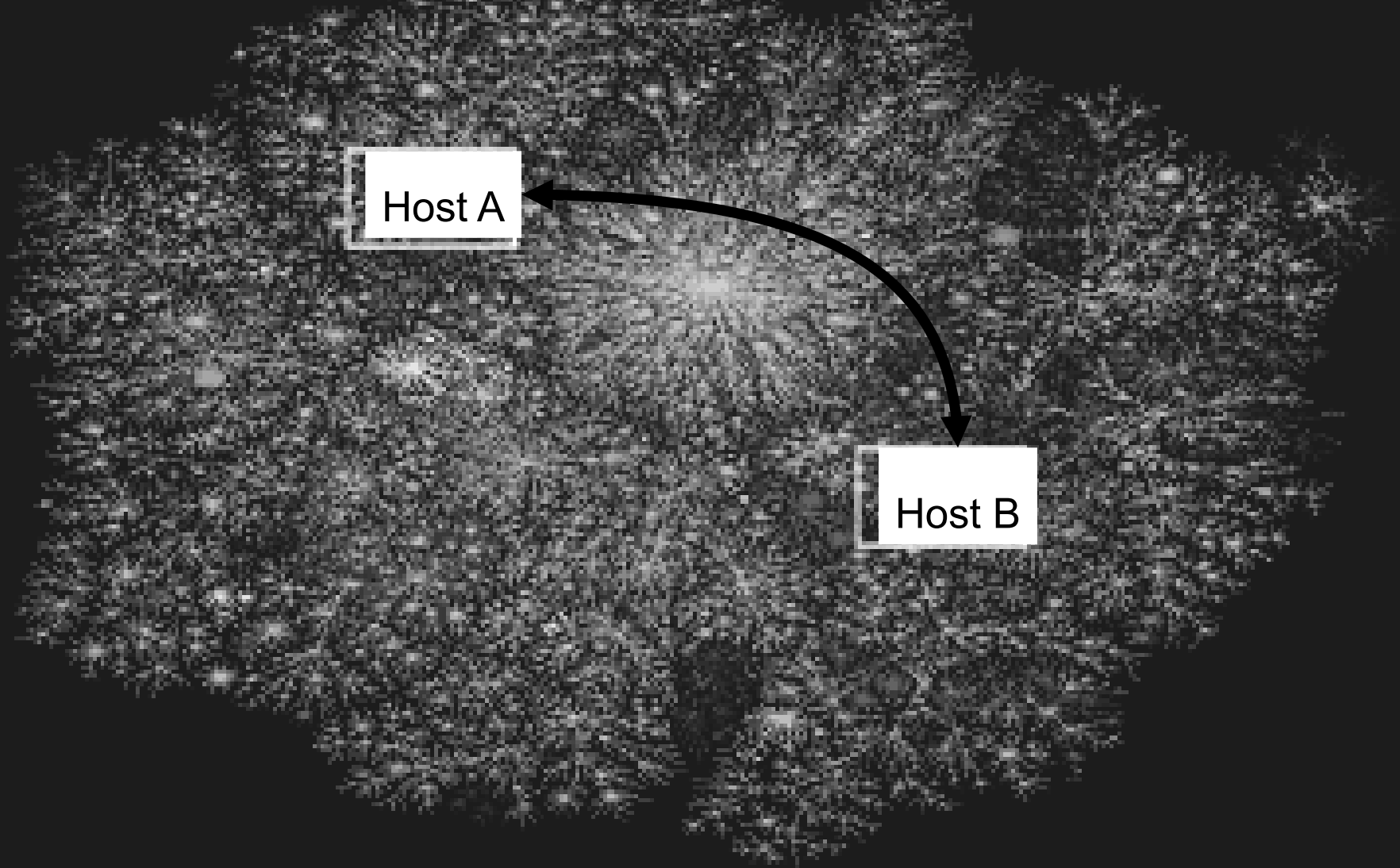
DAL PUNTO DI VISTA ORGANIZZATIVO:

*Un insieme di oltre 50000 **Autonomous Systems** alcuni su scala nazionale altri su scala continentale e intercontinentale*



Ma cos'è **INTERNET** (3)?

Un entità *trasparente* per gli host



In sintesi: cos'è Internet?

DAL PUNTO DI VISTA “FISICO”:

Un insieme di componenti interne (host, link, router) eterogenei, la complessità di gestire l'eterogeneità è delegata ai protocolli H2N, che devono essere “compatibili” con il protocollo IP ma possono essere progettati in modo indipendente

DAL PUNTO DI VISTA “FUNZIONALE”:

Rete globale organizzata in tante sotto-reti tramite un sistema di indirizzamento gerarchico, l'inoltro avviene mediante packet switching e inoltro stateless dei pacchetti, per ottenere massime prestazioni dai router

DAL PUNTO DI VISTA ORGANIZZATIVO:

Un insieme di organizzazioni e aziende (Autonomous Systems - AS), che amministrano in modo esclusivo e decentralizzato “porzioni di Internet” e collaborano per mantenere la rete globale e garantire un accesso “neutrale” agli utilizzatori

Alcuni principi funzionali di progetto

- **Survivability**
 - Se tra due host esiste un qualsiasi percorso, la comunicazione deve poter avvenire
- **Forma a clessidra su più livelli**
 - IP effettua minime assunzioni sui mezzi di trasporto sottostanti e deve funzionare per tutti i tipi di applicazioni di rete
- **Mancanza di “stato”**
 - La “intelligenza” è mantenuta ai bordi della rete (host) e non all’interno (router). Si facilitano la survivability del sistema e le prestazioni di trasmissione
- **Net neutrality**
 - Ogni pacchetto con qualsiasi mittente e destinazione è trattato nello stesso modo
 - Negli ultimi dieci anni tentativi/azioni di indebolimento e ripristino
- **Decentralizzazione organizzativa**
 - Ogni rete è potenzialmente posseduta e gestita da un ente diverso (AS)

Funzioni del livello 3 network (e scaletta degli argomenti da affrontare)

1. Si garantisce l'indirizzamento univoco degli host

- Tutti gli host collegati a Internet devono essere identificati ed in modo esclusivo → Indirizzo IP

2. Si definisce l'unità di trasferimento dati

- Definisce l'unità informativa utilizzata da Internet per trasferire dati → Datagram IP

3. Si chiarisce l'architettura di Internet

- Definisce i componenti fondamentali di una rete distribuita su scala geografica → Autonomous Systems, Router

4. Si illustrano le diverse funzioni di routing

- Gli algoritmi di routing determinano il percorso nell'ambito di una rete geografica attraverso il quale si consegnano i datagram
- Caratteristica **best-effort**: la consegna dei datagram è *non affidabile*

Indirizzi IP (IPv4)

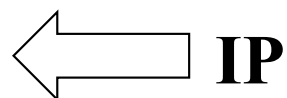
Indirizzi IP (IPv4)

- Un indirizzo IP ha dimensione 32 bit, ed è solitamente rappresentato tramite i valori di ciascuno dei 4 byte che lo compongono in notazione decimale
- Esempio: 155.185.121.7

Possibili scelte progettuali

- **Lunghezza indirizzi**

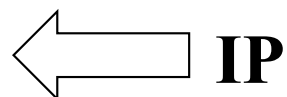
- lunghezza fissa
- lunghezza variabile



- Vantaggi a livello di flessibilità, ma maggiori costi nella gestione dei pacchetti e del routing

- **Spazio di indirizzamento**

- Gerarchico (strutturato)
- Flat



Indirizzi IP

- Per fornire un servizio di comunicazione universale (ogni nodo della rete può comunicare con ciascun altro nodo) occorre un metodo che permetta di identificare univocamente ogni nodo
 - A ogni nodo è assegnato un unico indirizzo Internet (indirizzo IP) formato da 32 bit $\rightarrow 2^32 \cong 4,3$ miliardi di indirizzi diversi
- L'indirizzo IP (32 bit) è suddiviso in 4 campi:
 - Ciascun campo è formato da un byte (8 bit)
 - E' separato da un punto (notazione decimale puntata o ***dotted notation***)
 - Esempio: **130.192.5.189**

Componenti dell'indirizzo IP

Ogni indirizzo IP è solitamente strutturato in una coppia:

<netid, hostid>

dove **netid** (o prefisso di rete) identifica la rete
e **hostid** identifica un host di quella rete

**Questa notazione consente di indicare sinteticamente
intervalli contigui di indirizzi**

**(anche detti “blocchi” o “range” di indirizzi
definiti da “prefissi di rete”)**

Componenti dell'indirizzo IP

Ad esempio: **128.211.121.7**

L'indirizzo può essere composto da due parti

NetId: 128.211 HostId: 121.7

Cosa determina le parte di NetId e di HostId?

Dipende:

- Assegnazione di **classi predefinite**
- Ripartizione manuale tramite notazioni **classless**

Ricordare l'obiettivo: ridurre il numero di regole di routing da memorizzare nelle tabelle di routing

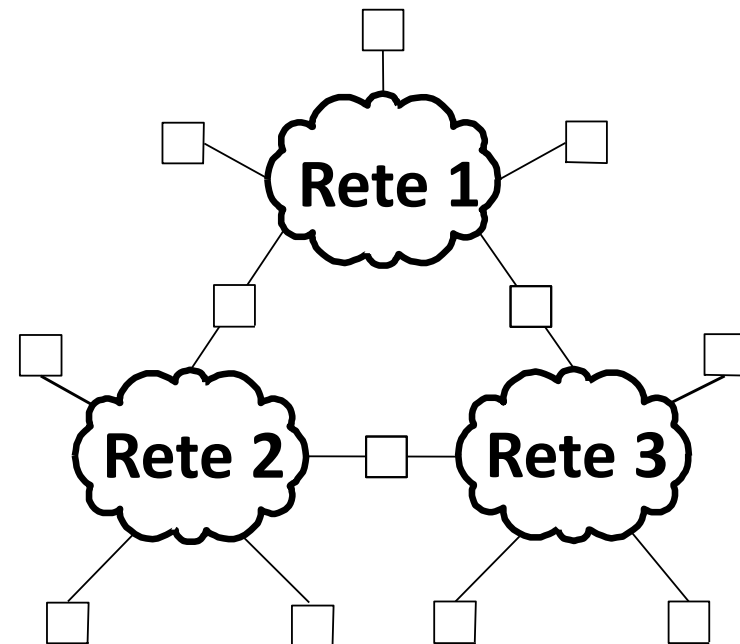
Assegnazione indirizzi IP nelle reti [1]

- Lo spazio degli indirizzi di IP viene *solitamente* gestito in «blocchi» (o «intervalli») di indirizzi

NetId Rete 1: 128.211

NetId Rete 2: 128.212

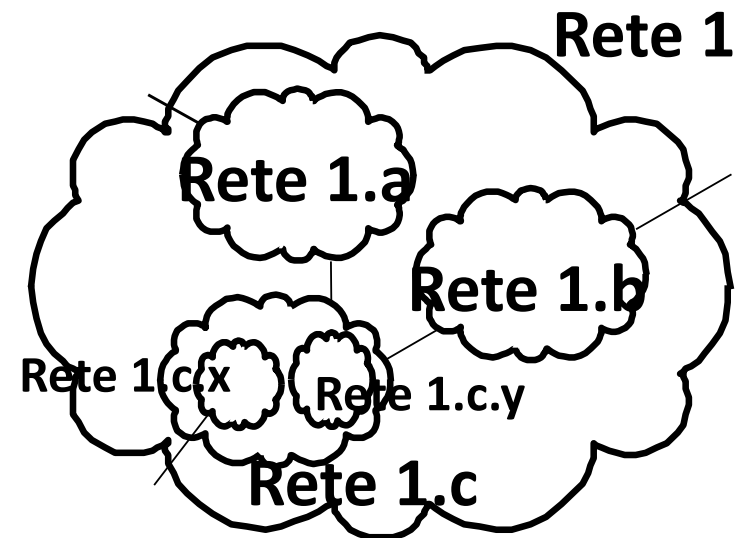
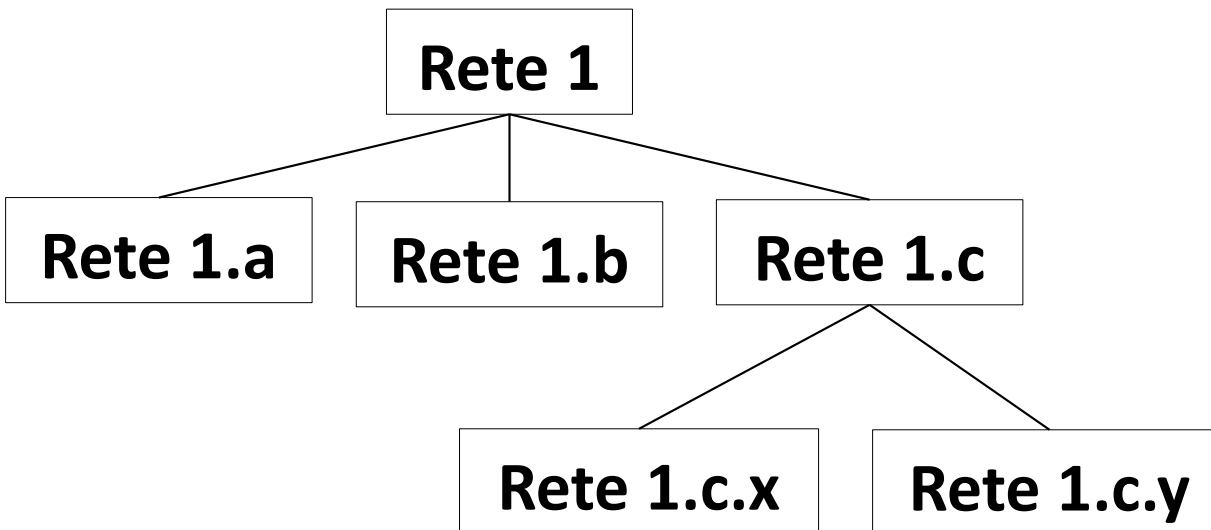
NetId Rete 3: 128.213



- Ogni rete gestisce in modo esclusivo tutti gli indirizzi IP sottintesi dal NetId assegnato
 - Univocità di indirizzi a livello di rete

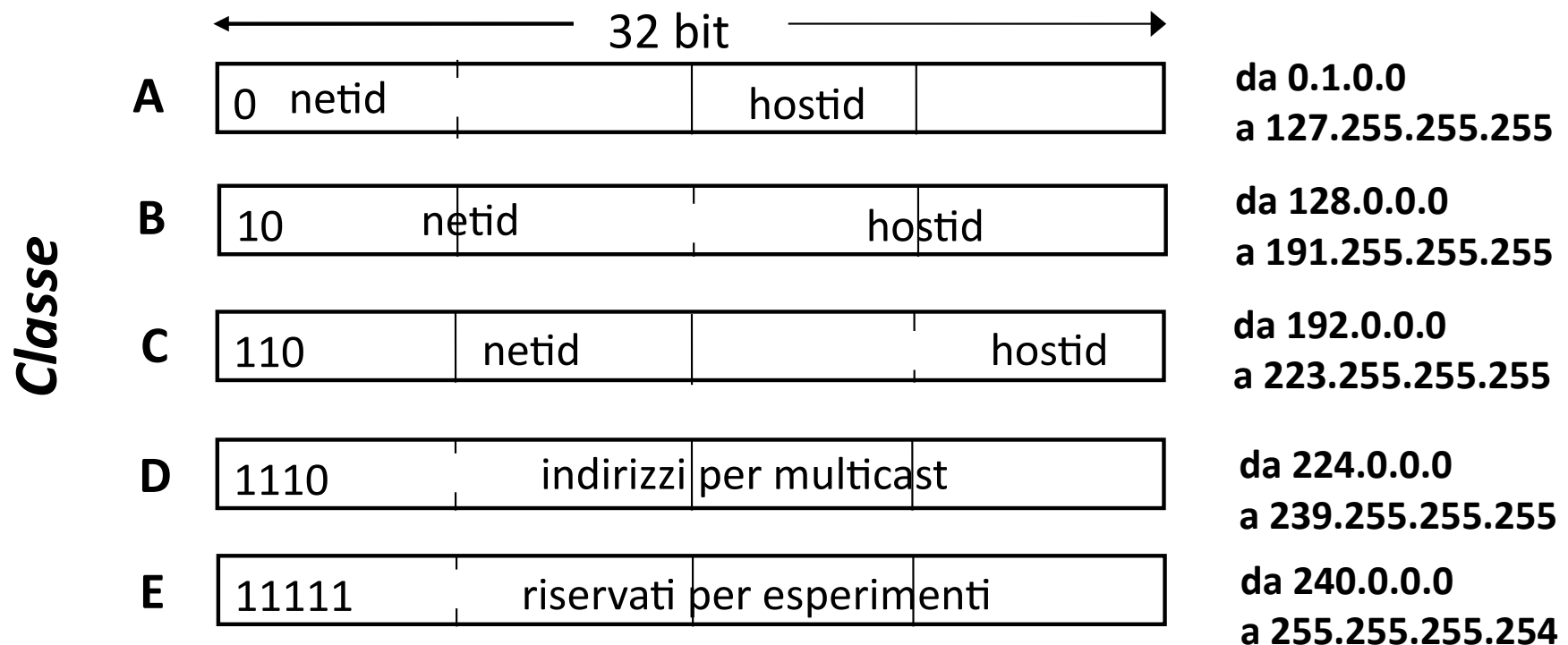
Assegnazione indirizzi IP nelle reti [2]

- L'approccio è **ricorsivo** (consistente con la definizione di rete)
 - Un blocco di indirizzi può essere assegnato a una rete locale in cui tutti gli indirizzi possono comunicare a livello H2N
 - Oppure essere assegnato a una rete «logica» che contiene più reti («logiche» o locali)
 - In questo caso si parla di **subnetting** (approfondimenti dopo)



Classi di indirizzi IP

- 3 classi utilizzabili per l'indirizzamento di host (*classe A*, *classe B*, *classe C*), più 1 classe per multicast address (*classe D*), più 1 classe riservata (*classe E*)
 - La quantità di bit destinati al prefisso di rete dipende dalla classe cui l'indirizzo appartiene
 - La classe è codificata dai bit più significativi dell'indirizzo



Dimensioni delle classi di indirizzi

- **Classe A** (7 bit per netid, 24 bit per hostid)
 - 128 (**2**) possibili network ID
 - Oltre 16 milioni di host ID per ciascun network ID
- **Classe B** (14 bit per netid, 16 bit per hostid)
 - 16K = 16384 (**2**) possibili network ID
 - 64K = 65536 (**2**) host ID
- **Classe C** (21 bit per netid, 8 bit per hostid)
 - Oltre 2 milioni (**2**) di possibili network ID
 - 256 (**2⁸**) host ID

Indirizzi *classless* (CIDR)

- Oggi sono necessarie architetture più flessibili
- Gli indirizzi non sono considerati in classi fisse, ma l'intero spazio di indirizzamento può essere suddiviso in blocchi di dimensioni differenti
- Si usa la notazione **CIDR (Classless Inter-Domain Routing)** dove ciascun insieme di bit del netid è indicato dal suffisso ***n*** nella notazione

a.b.c.d/n

Indirizzi *classless*

- La ripartizione degli indirizzi in classi è molto rigida e poco graduale perché basata su interi byte:
 - si passa da reti con 250 host (Classe C) a reti con 65000 host (Classe B) a reti con milioni di nodi (classe A)
- Per motivi gestionali e di efficienza del routing interno, può convenire definire degli “insiemi logici” di indirizzi più flessibili rispetto alla suddivisione rigida in 1, 2, 3 byte per il *netid*
- Più flessibili significa passare *da una suddivisione in byte ad una suddivisione in bit* per la coppia *<netid, hostid>*

Indirizzi Classless (CIDR)

- Con la notazione CIDR si elimina il concetto di indirizzamento a classi fisse A-B-C-D: l'indirizzo IP non ha più un confine fisso tra netid e hostid
- Si utilizza la notazione slash per indicare il numero di bit usati per netid **a.b.c.d/x**. Es., 197.8.3.0/24
- Gli indirizzi CIDR richiedono l'utilizzo di strutture dati e algoritmi opportuni da utilizzare per consultare in modo efficace le **tabelle di routing**
 - **Le tabelle di routing devono conservare anche l'informazione relativa alla netmask, e non solo la rete di destinazione**
- Si utilizza un approccio di tipo **longest prefix** per gestire possibili conflitti fra diverse regole di routing

Approfondiremo in seguito

Assegnamento indirizzi IP

- **Gli indirizzi IP sono indirizzi *logici* (non fisici)**
- Ciascun host deve essere identificato da un indirizzo IP, che può essere assegnato:
 - permanentemente ad un host
 - oppure dinamicamente al momento del boot di un host
- **Come fa un host a conoscere il proprio indirizzo IP?**
 - **Configurazione manuale:** l'indirizzo IP è configurato in un file dall'amministratore del sistema
 - **Dynamic Host Configuration Protocol (DHCP):** allocazione dinamica effettuata da un server speciale

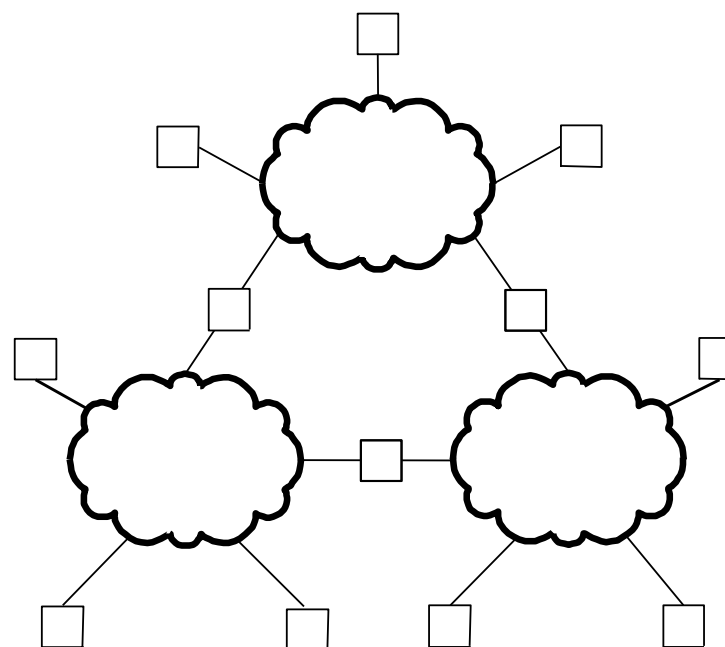
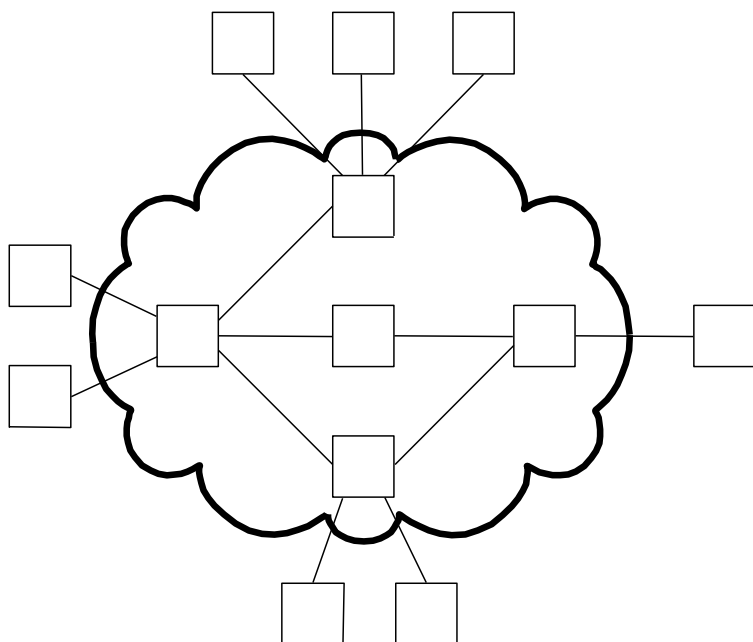
Indirizzi IP speciali

- **Network address:** **hostid** con tutti i bit uguali a 0 (es., 128.211.0.0 indica la rete di classe B avente netid 128.211) → denota il **netid** (prefisso) assegnato ad una rete
- **Directed broadcast address:** **hostid con tutti i bit uguali a 1** (es., 128.211.255.255 indica il broadcast per la rete di classe B avente netid 128.211) → permette il broadcast a tutti gli host di una certa rete
- **Limited broadcast address:** **tutti i bit uguali a 1** (ossia 255.255.255.255) → permette il broadcast sulla rete fisica locale
- **Nessun indirizzo IP:** **tutti i bit uguali a 0** (ossia 0.0.0.0) → usato per il boot dell'host o per configurazioni «particolari» (es: bridge, sniffing “stealth”)
- **Loopback address (localhost):** la classe A con netid pari a 127 (es., 127.0.0.1) → è un indirizzo software virtuale senza corrispettivo hardware e senza connessioni di rete: è usato per il testing di applicazioni di rete (ad es., consente di comunicare con un server sulla stessa macchina: <http://127.0.0.1>)

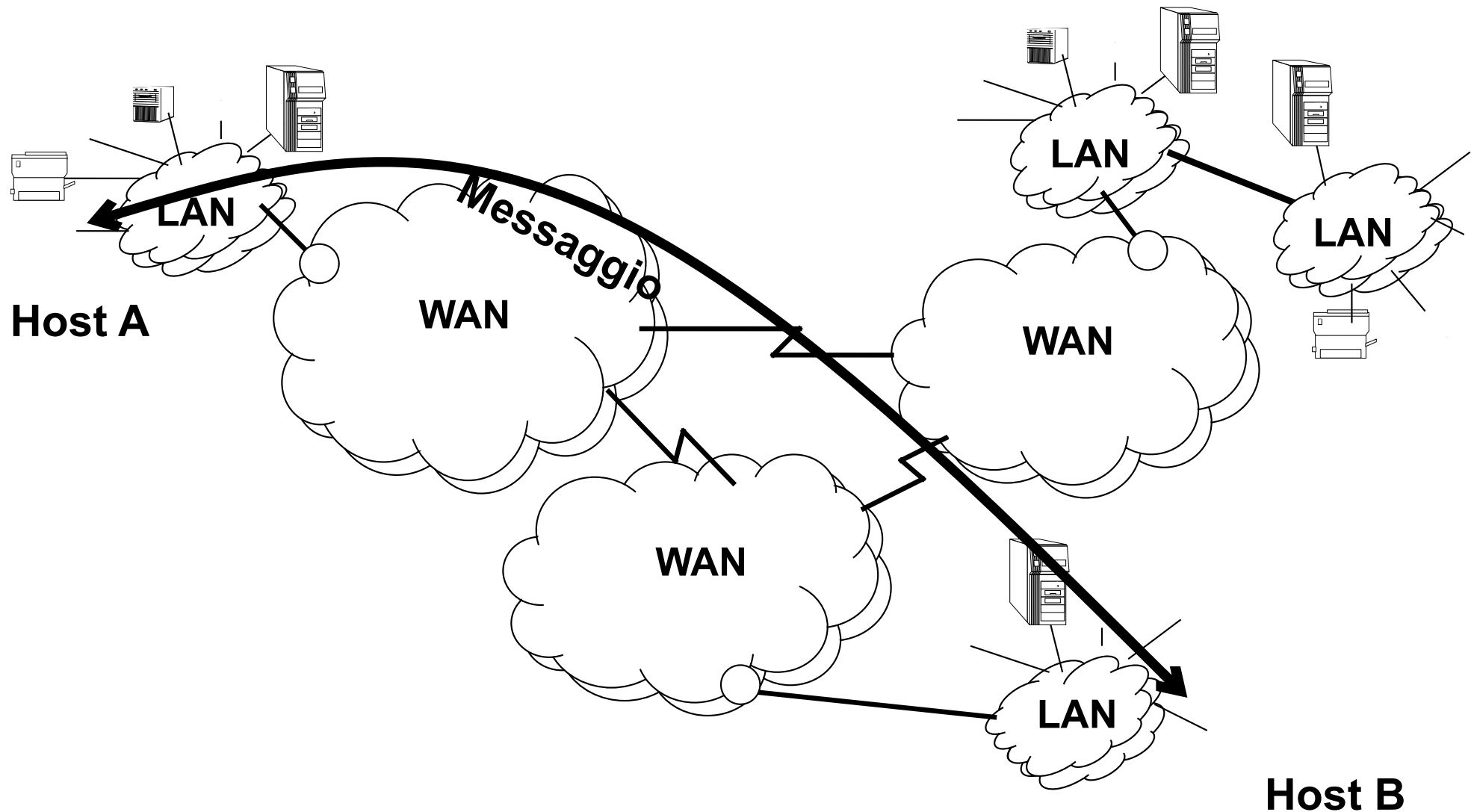
Concetti di instradamento dei pacchetti IP

Modellazione ideale reti

- Una rete può essere definita ricorsivamente
 - Due o più nodi connessi tramite collegamenti
 - Due o più reti connesse tramite nodi



Comunicazione logica tra due host

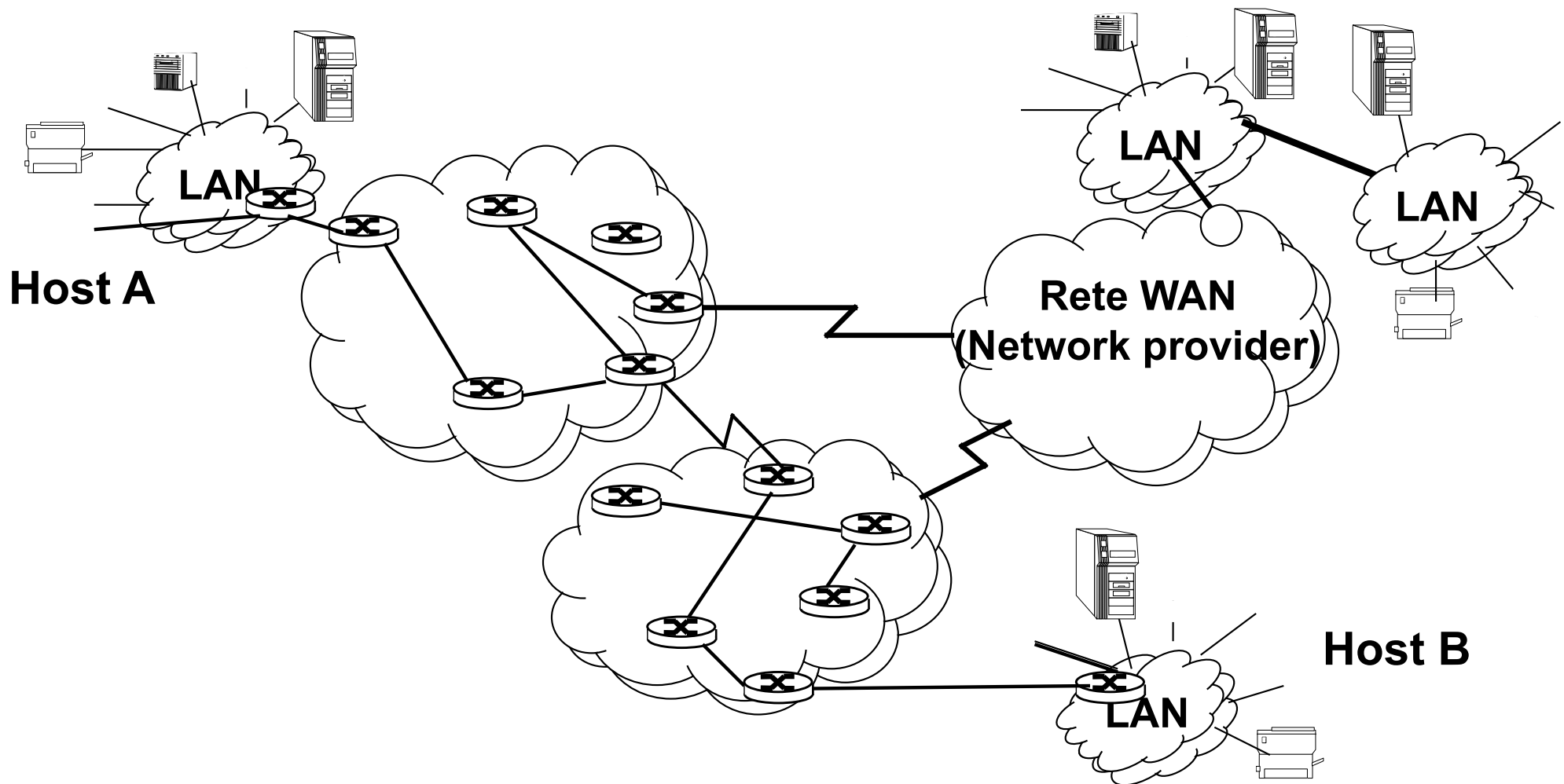


Logicamente comunicano i due host terminali

Comunicazione reale

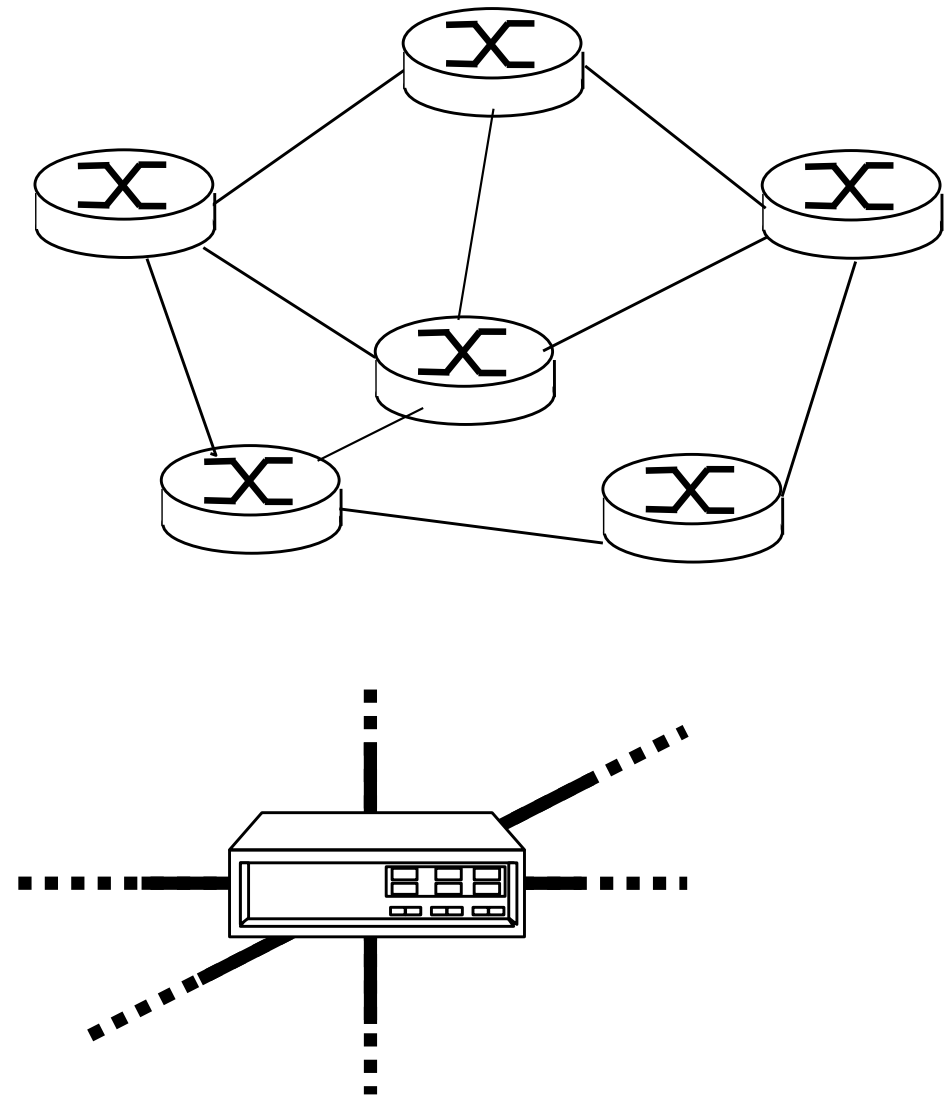
In realtà, le informazioni attraversano tutti i nodi e i collegamenti

- Problemi di instradamento e di *condivisione delle risorse*

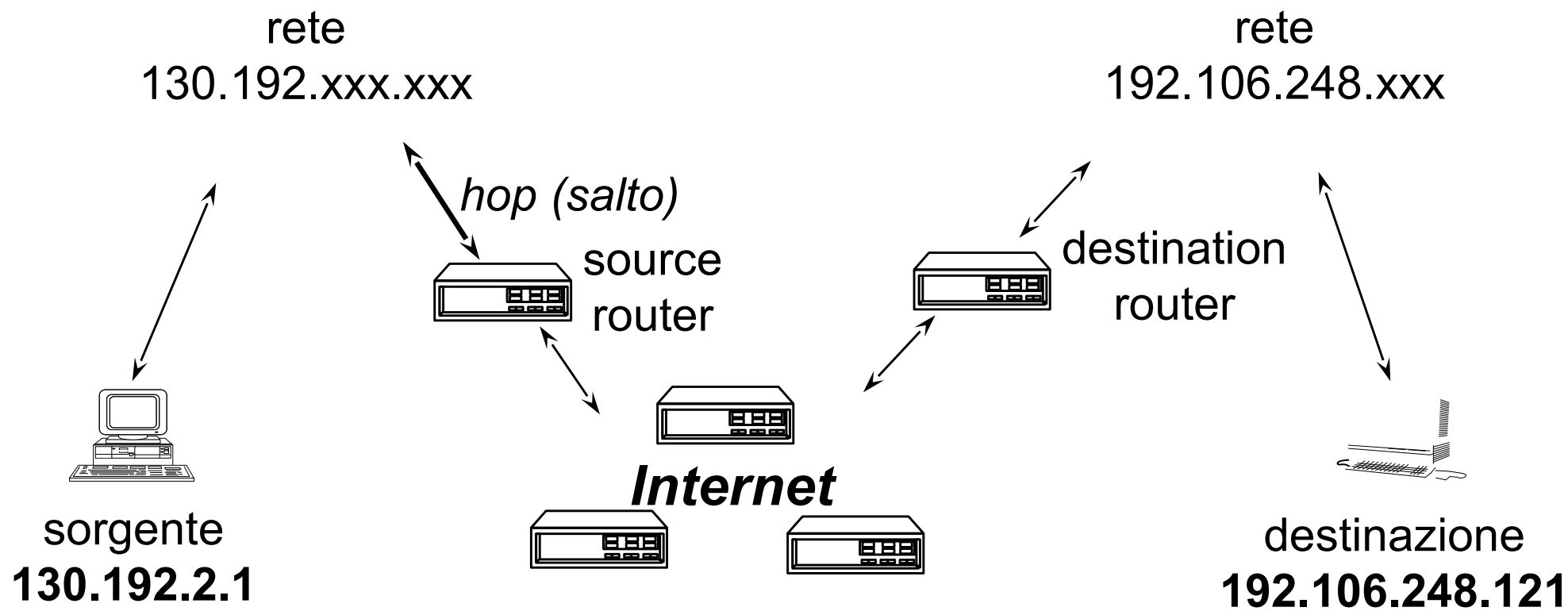


Router

Il router deve risolvere un problema molto ben definito: Instradare i pacchetti nella rete da un qualsiasi host ad un qualsiasi altro host, sulla base dell'indirizzo IP destinazione incluso nel pacchetto stesso



Routing IP



- I router si passano i pacchetti hop-by-hop: non si decide il percorso complessivo, ma solo il router successivo
- A volte il routing non ha successo perché i router sovraccarichi scartano pacchetti (**congestione**, **limite fisico**) o vi possono errori di routing (**errore logico**, **ad esempio cicli nella rete**) – vedremo in seguito

Inoltro hop-by-hop dei pacchetti IP

- Un host che invia un pacchetto all'esterno della propria rete locale deve decidere tramite router inviarlo: questo router viene detto ***first hop router*** o ***source router***
- Ogni router deve decidere a sua volta il router (***next-hop router***) a cui inoltrare il pacchetto
- Infine, il pacchetto dovrebbe arrivare all'host destinazione
 - Se il pacchetto percorre troppi router, potrebbe essere scartato (si vedrà il TTL nell'header IP)
 - L'ultimo router prima dell'arrivo del pacchetto a destinazione è anche detto ***destination router***

Problema del routing

- Consegna i pacchetti da un host sorgente a una destinazione potenzialmente attraversando molteplici router intermedi
 - in modo best effort, privo di connessione, e quindi non garantito
- Quando un problema è complesso si suddivide in sottoproblemi più semplici:
 - **Sottoproblema 1**: ad ogni pacchetto in ingresso, determinare il link di uscita in modo che il pacchetto si avvicini alla destinazione (**IP forwarding**)
 - **Sottoproblema 2**: mantenere **informazioni aggiornate** per risolvere il sottoproblema 1 (**protocollo di routing**)

IP Forwarding

- **IP forwarding (inoltro)**: meccanismo con cui un router trasferisce i datagram da un'interfaccia d'ingresso a quella in uscita
- Effettuato da ogni router
- Il **next-hop router** appartiene a una rete alla quale il **router è collegato a livello H2N**

Per inoltrare i pacchetti:

- l'indirizzo di destinazione viene estratto dall'header del datagram (*prossime slide*)
- l'indirizzo di destinazione è usato come indice nella **tabella di routing** (*prossime lezioni*)

Caratteristiche dell'IP forwarding

- **Indipendenza dal mittente:** il next-hop routing, tipicamente, non dipende dal mittente del pacchetto o dal cammino che il pacchetto ha attraversato fino a quel momento
 - **Il router estrae dal pacchetto soltanto l'indirizzo del destinatario**
- La tabella di routing deve contenere un next-hop router per ciascuna destinazione
- **Il next-hop router appartiene a una rete alla quale il router è collegato direttamente**

Tabella di routing [1]

- Ogni host e ogni router hanno una tabella di routing in cui ciascuna riga fornisce il next-hop per ogni possibile destinazione
 - Il percorso dei pacchetti viene selezionato hop-by-hop

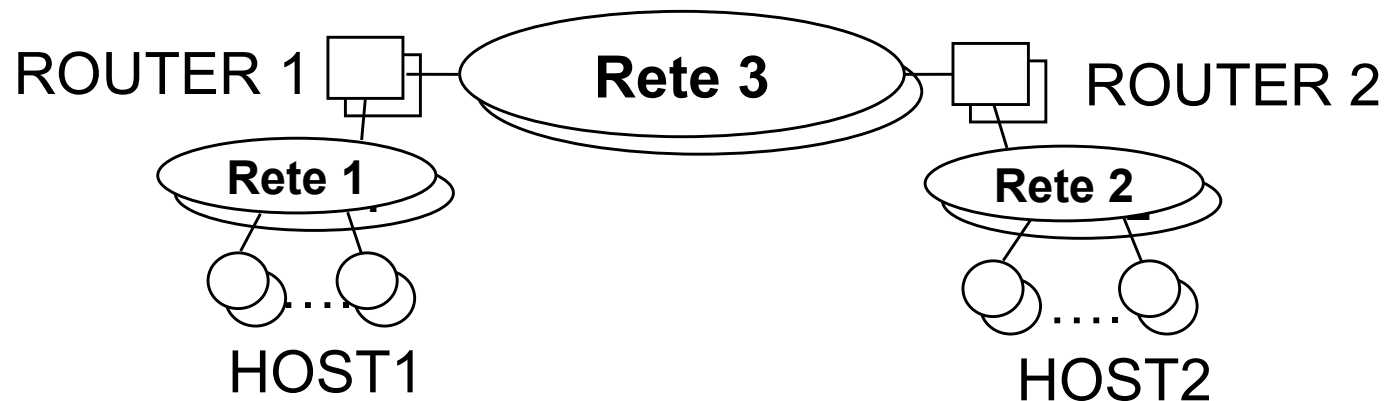


Tabella di routing host Rete 1

Destinazione	Metodo
Rete 1	H2N
Rete 2	IP tramite Router 1
Rete 3	IP tramite Router 1

Tabella di routing host Rete 2

Destinazione	Metodo
Rete 1	IP tramite Router 2
Rete 2	H2N
Rete 3	IP tramite Router 2

Tabella di routing [2]

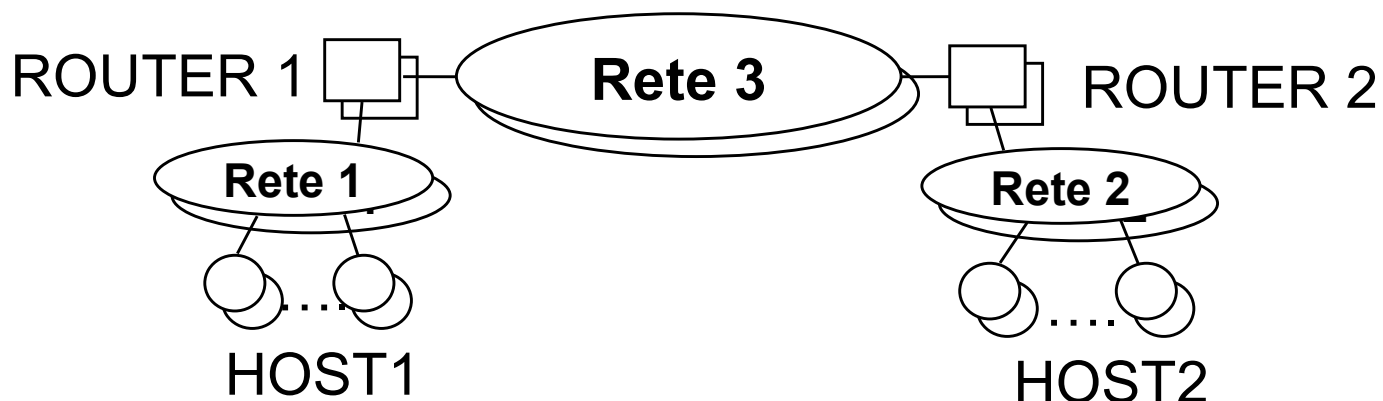


Tabella di routing host Rete 3

Destinazione	Metodo
Rete 1	IP tramite Router 1
Rete 2	IP tramite Router 2
Rete 3	H2N

Tabella di routing host Router 1

Destinazione	Metodo
Rete 1	IP tramite Router 1
Rete 2	IP tramite Router 2
Rete 3	H2N

Tabella di routing host Router 2

Destinazione	Metodo
Rete 1	IP tramite Router 1
Rete 2	H2N
Rete 3	H2N

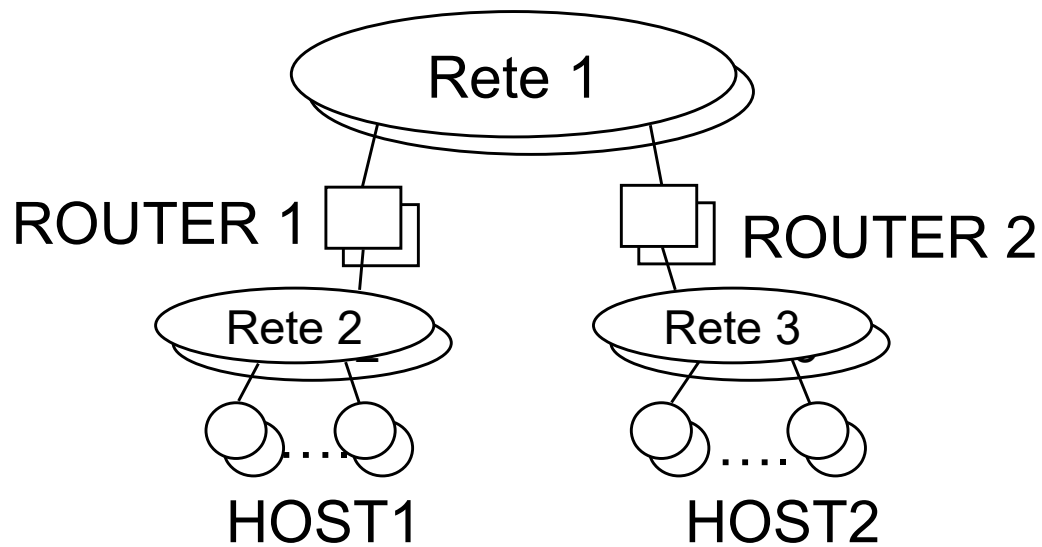
Funzionamento del router

Ogni router l'indirizzo IP di destinazione dall'header IP e consulta la tabella di routing per determinare:

1. Se l'indirizzo appartiene a una rete nota a cui il router è connesso a livello H2N, viene usato il protocollo H2N per inviare il pacchetto a D
 - Se Ethernet, risoluzione indirizzo HW con ARP e costruzione frame verso D)
2. Se l'indirizzo appartiene a una rete nota a cui il router non è connesso a livello H2N, nella tabella di routing è presente il **next-hop router** a cui inviare il pacchetto
 - Comunico a livello H2N con il next-router, e a livello IP con D
3. Se l'indirizzo non appartiene ad alcuna rete nota, ma esiste un **router di default (default gateway)**, si invia il pacchetto al router in modo analogo a come descritto nel punto 2
4. Altrimenti, non invio il pacchetto (tipico errore **network unreachable**)

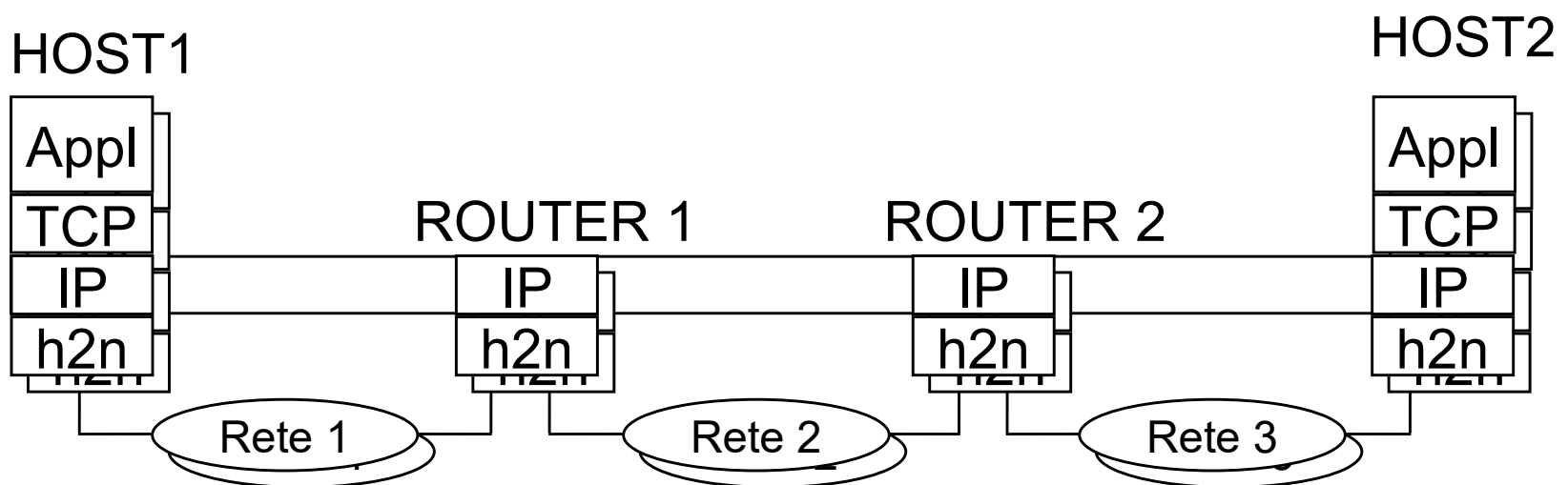
Nota: nel caso nella tabella di routing esistano più reti a cui la destinazione può far riferimento, si usa la regola del “prefisso più lungo” (longest prefix) – **discusso in laboratorio**

Distinguere i due casi fondamentali



Host mittente e destinatario sono nella stessa sottorete

Host mittente e destinatario sono in sottoreti differenti



Dimensioni tabella di routing

- Le dimensioni (crescenti) delle tabelle di routing potrebbero essere un limite allo sviluppo di Internet
 - Abbiamo già detto che l'uso di indirizzi gerarchici serve a evitare indirizzamento per singoli IP
 - Possiamo però aggregare
- Si sfruttano **tecniche di aggregazione** per fare in modo che ogni riga possa “catturare” molte reti di destinazione
 - Essenziale **progettare le reti IP assegnando opportunamente gli indirizzi IP** (e.g., indirizzi adiacenti per una stessa rete locale)
 - Essenziale utilizzare indirizzamento gerarchico e organizzazione appropriata di subnetting e supernetting (prossimo argomento)

Protocolli di routing

- In contesti semplici, la tabella di routing può essere definita in maniera **statica** da un amministratore di rete o da un protocollo di configurazione (e.g., DHCP)
 - Ad esempio, tabella di un host o router di reti locali molto semplici
- I **protocolli di routing** (e.g., **RIP**, **OSPF**, **BGP**) servono invece a costruire dinamicamente le tabelle di routing presenti sui router
 - Capacità di popolare in modo «ottimale» la tabella in base dalla topologia della rete e delle sue caratteristiche
 - Capacità di **adattamento** a fronte di **guasti** o, potenzialmente, **congestioni**
 - In reti mediamente complesse, l'intervento manuale umano per definire e aggiornare configurazioni statiche non è accettabile
- **Approfondiremo in seguito**

Subnetting e Supernetting (IPv4)

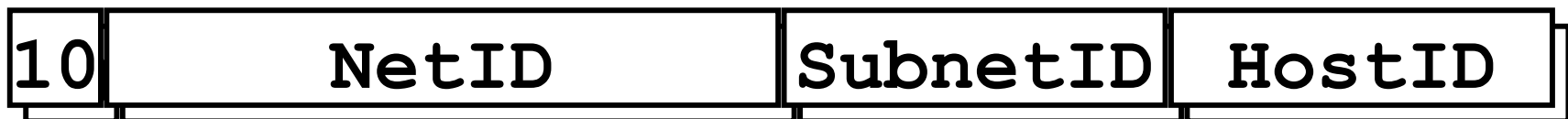
(complementare con esercizi “laboratorio”)

Subnetting e Supernetting

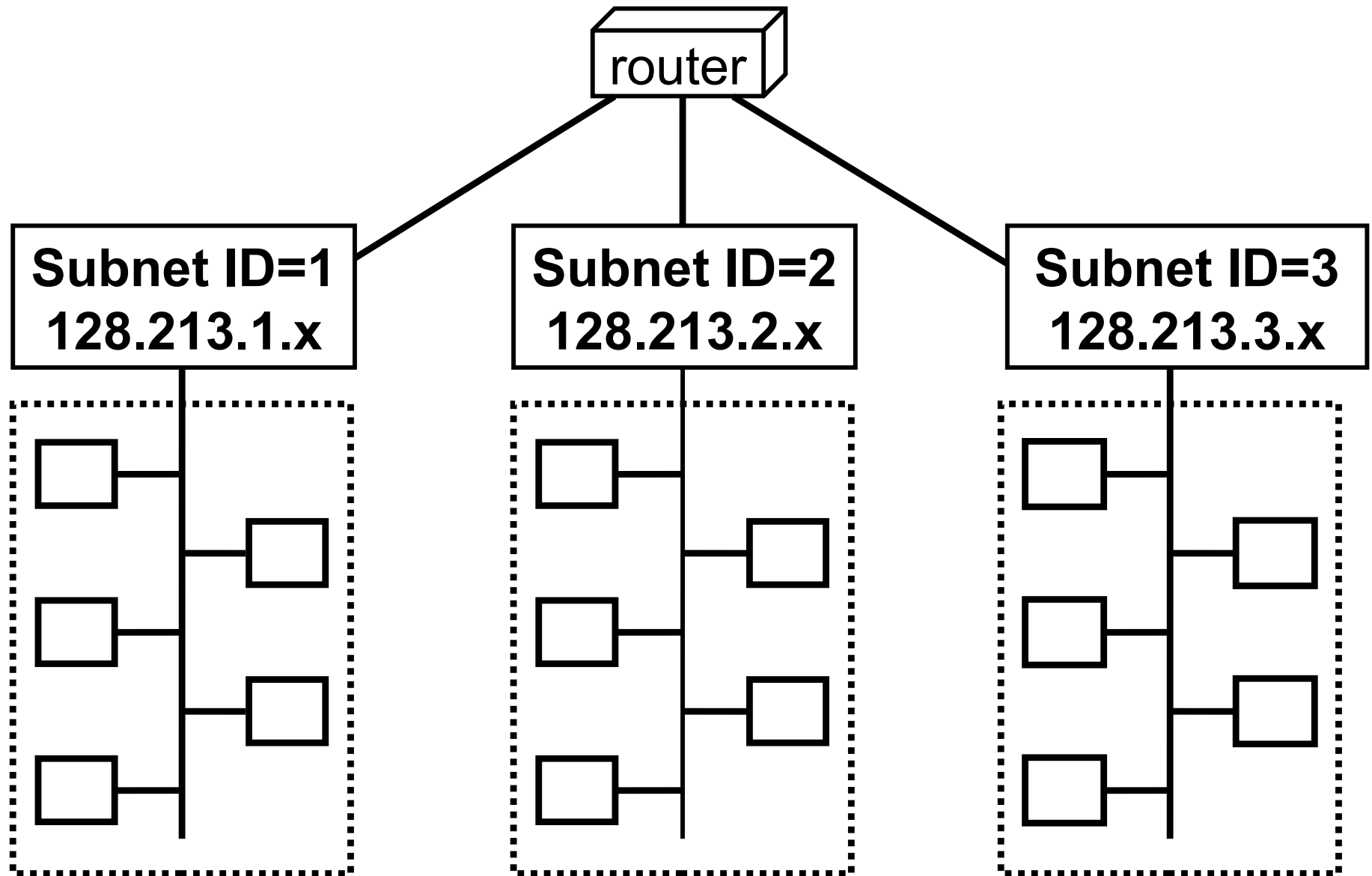
- Due opportunità:
 - sottoclassi di indirizzi IP (**subnet**), soprattutto per organizzazioni con indirizzi di classe B
 - sopraclassi di indirizzi IP (**supernet**), per organizzazioni grandi con più indirizzi di classe C ovvero per ISP
- Due vantaggi:
 - Si crea maggiore flessibilità nella ripartizione degli indirizzi all'interno di un'organizzazione (es., Università con indirizzi di Classe B)
 - Si facilitano le operazioni di routing dei pacchetti identificando insiemi di indirizzi di host contigui

Subnetting

- Un'organizzazione può suddividere il suo spazio di host address in gruppi detti **subnet**
- Il **subnet ID** è tipicamente utilizzato per raggruppare host basati sulla topologia fisica della rete
- Per esempio, per un indirizzo di classe B, si può avere:

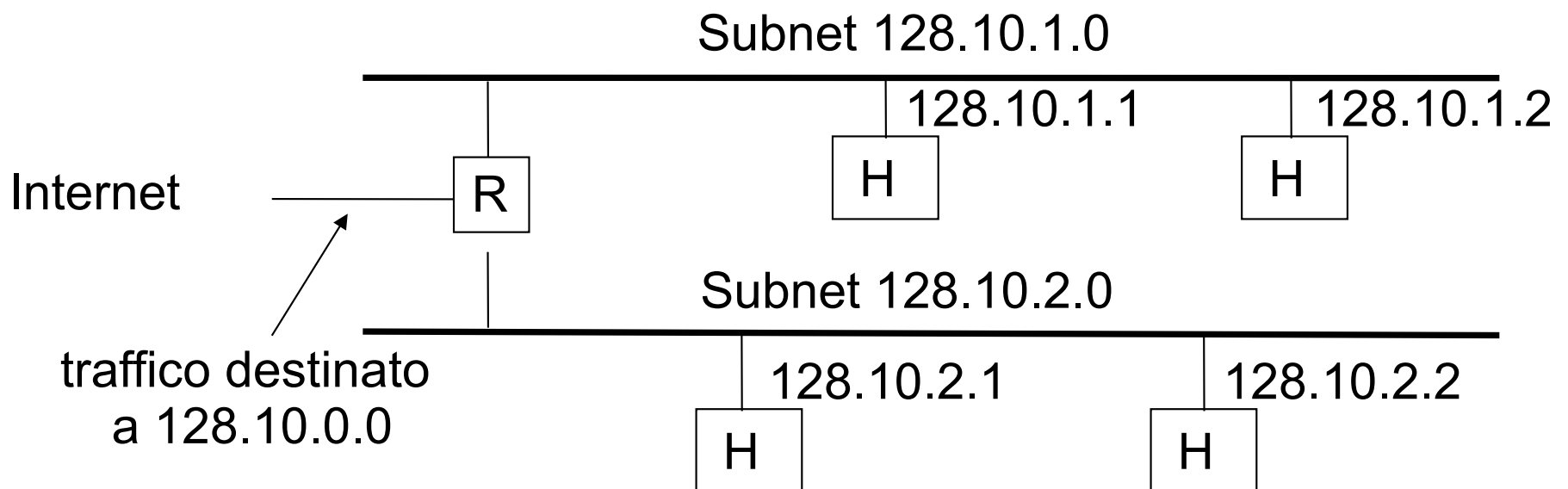


Subnetting (*cont.*)



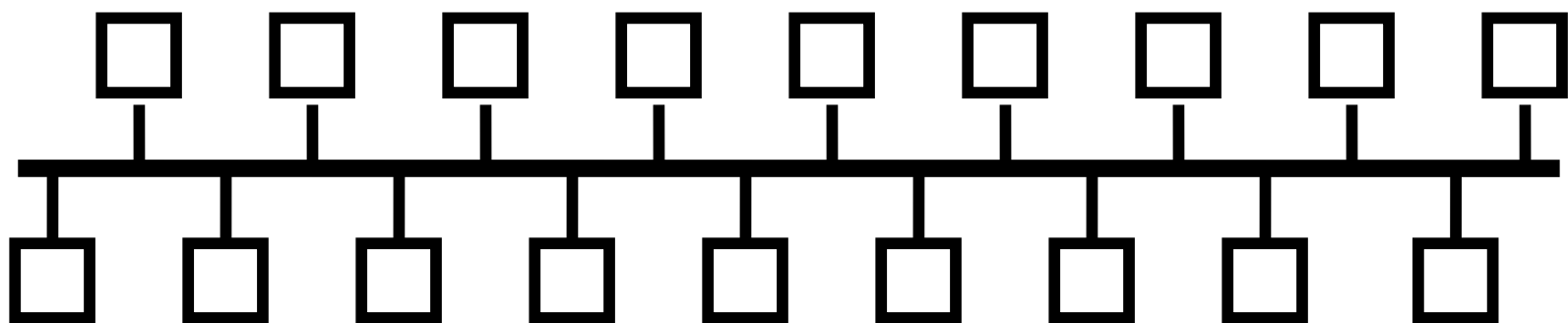
Subnet addressing

- **Schema di indirizzamento IP originale:**
 - ad ogni rete fisica è assegnato un unico “indirizzo di sottorete”
 - ogni host appartenente a questa rete ha come netid l’indirizzo di sottorete



Subnetting addressing (*cont.*)

- Il subnetting consente la massima flessibilità
- E' possibile anche avere uno stesso segmento di rete fisico suddiviso in multiple subnet logiche, corrispondenti per esempio a diversi gruppi di una organizzazione
- Es.,



Uso della network mask

- Per definire i bit (non i byte!) dedicati al *netid* si usa una *network mask* di 4 byte. Es.

Net mask: 11111111.11111111.11111111.11000000

- La *network mask* permette di individuare due dati mediante un AND logico con l'indirizzo IP:
 - quale parte di un indirizzo IP è riservata per il *netid* (la parte di 1)
 - quale parte è disponibile per gli *hostid* (la parte di 0)

Esempi di subnet mask

- Implementazione delle subnet usando le maschere:
 - **subnet mask** formata da 32 bit per ciascuna rete che usa il subnet addressing
 - nella mask, i bit settati ad 1 corrispondono alla parte di rete, quelli settati a 0 alla parte host
- Esempio di rete di classe B con cinque reti fisiche suddivise su tre livelli:
 - maschera = **11111111 11111111 11100000 00000000**
- Esempio di rete in cui tutto il terzo byte dell'indirizzo IP è usato per la subnet:
 - maschera = **11111111 11111111 11111111 00000000**

Subnet mask: esempio di uso

- Indirizzo IP: **156.154.81.56**
- Network mask: **255.255.255.240**
- A quale sottorete appartiene?

Indirizzo IP: 10011100.10011010.01010001.00111000

Subnet mask: 11111111.11111111.11111111.11110000

(AND) -----

Subnet: 10011100.10011010.01010001.00110000

- Qual è il range di host della sottorete?
 - Ci sono 2^{-2} host nella subnet, dove n è il numero degli ultimi 0 della subnet mask. Nell'esempio: $2^{-2}=14$, ovvero da 156.154.81.49 a 156.154.81.62
- Qual è il *broadcast address* della sottorete?
 - 10011100.10011010.01010001.00111111 → 156.154.81.63

Esempio di subnetting

Contesto

- Una università con un indirizzo di classe B: **150.100**
- Si assuma che ciascun dipartimento abbia meno di 100 host
- Quanti bit servono per identificare gli host di una sottorete?

7
- Qual è la network mask?
 - **11111111 11111111 11111111 10000000**
 - **255.255.255.128**

Esempio di subnetting (*cont.*)

network	host
---------	------

network	subnet	host
---------	--------	------

1111...	...1111	10000000
---------	---------	----------

mask

Come usare le subnet mask per routing

- Le subnet servono soprattutto per facilitare il routing dei pacchetti all'interno della rete amministrata
- Si assuma, nel caso dell'università precedente, che arrivi un pacchetto con indirizzo destinazione: **150.100.12.176**
- Si effettua un AND tra l'indirizzo e la subnet mask
 - (150.100.12.176) *AND* (255.255.255.128)
 - Risultato: **150.100.12.128** che corrisponde alla sottorete di destinazione i cui host si trovano nel range **150.100.12.129 - 150.100.12.254**

Subnet addressing

Il **subnet addressing** modifica l'interpretazione degli indirizzi IP: l'indirizzo IP è composto da una porzione di **rete** e una **locale**

rete	locale	
rete	rete fisica	host

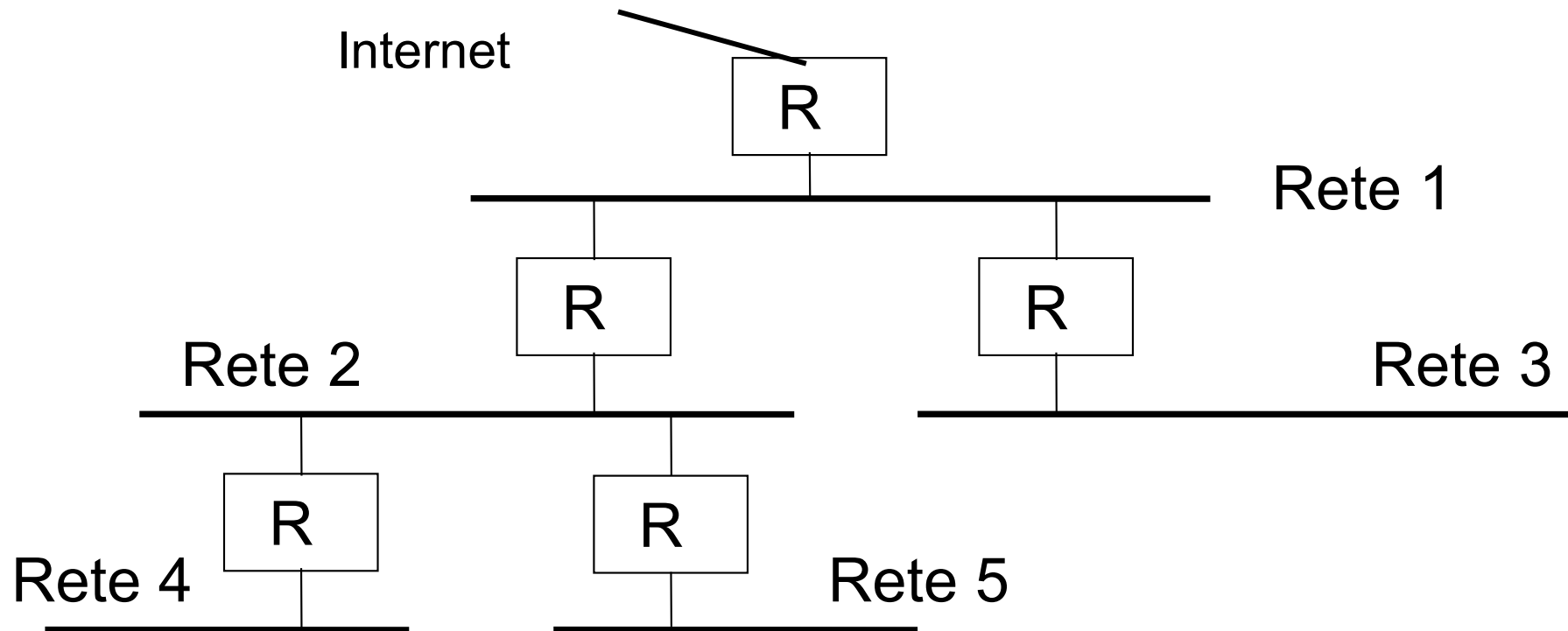
Risultato: *indirizzamento gerarchico* → **routing gerarchico**

Es. routing gerarchico: i router esterni usano i primi due byte dell'indirizzo IP per il routing, mentre il router della rete locale usa il terzo byte dell'indirizzo IP

Esempio di subnet gerarchico

Esempio di rete con cinque reti fisiche suddivise in tre livelli:

- rete di classe B (16 bit per parte locale)
- 5 reti fisiche: occorrono 3 bit (essendo $5 < 2^3 = 8$) per identificarle
- ad ognuna delle 5 reti fisiche è possibile collegare: $2^{16} = 65536$ host



Supernet

- **PROBLEMA** → Esaurimento dello spazio di indirizzamento all'interno di una stessa classe di indirizzi (B o C)
- **SOLUZIONE** → Un'organizzazione può richiedere più indirizzi della stessa classe per la sua rete. Es.
 - un blocco di indirizzi di classe C contigui viene assegnato a un'organizzazione
 - un blocco di indirizzi di classe B contigui viene assegnato a un Internet Service Provider
- **Come gestirli?**

Supernet (*cont.*)

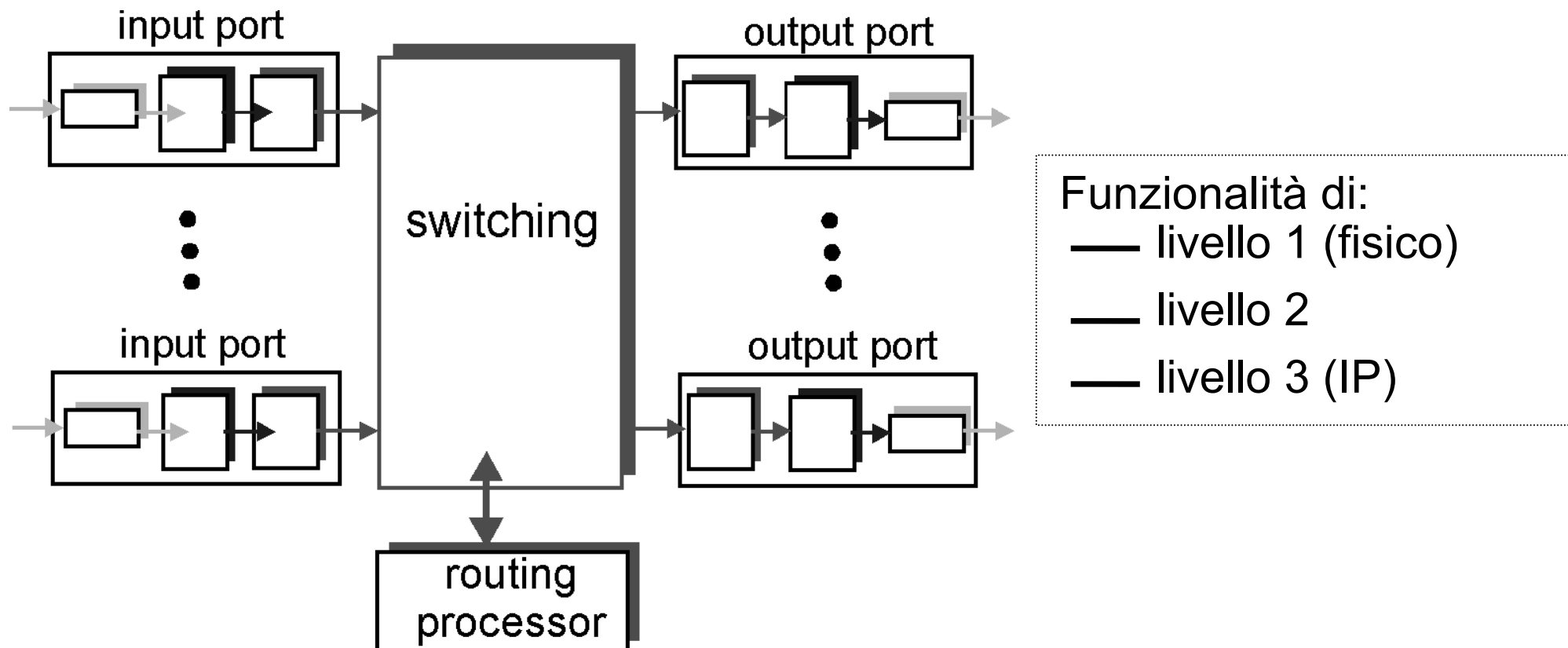
- **Gestione mediante supernet addressing:**
 - approccio opposto al subnet addressing
 - in pratica, si utilizzano meno bit di un intero byte per identificare il netid
- Formalmente, nei router si utilizza il meccanismo di **Classless Inter-Domain Routing (CIDR)** in cui:
(network address, count)
 - **network address** è il più piccolo indirizzo (in bit) nel blocco di indirizzi di classe B o C assegnati
 - **count** è il numero di blocchi di indirizzi di classe B o C contigui

Cenni di architetture di router

Cenni di architetture di router [1]

4 componenti fondamentali nell'architettura di un router:

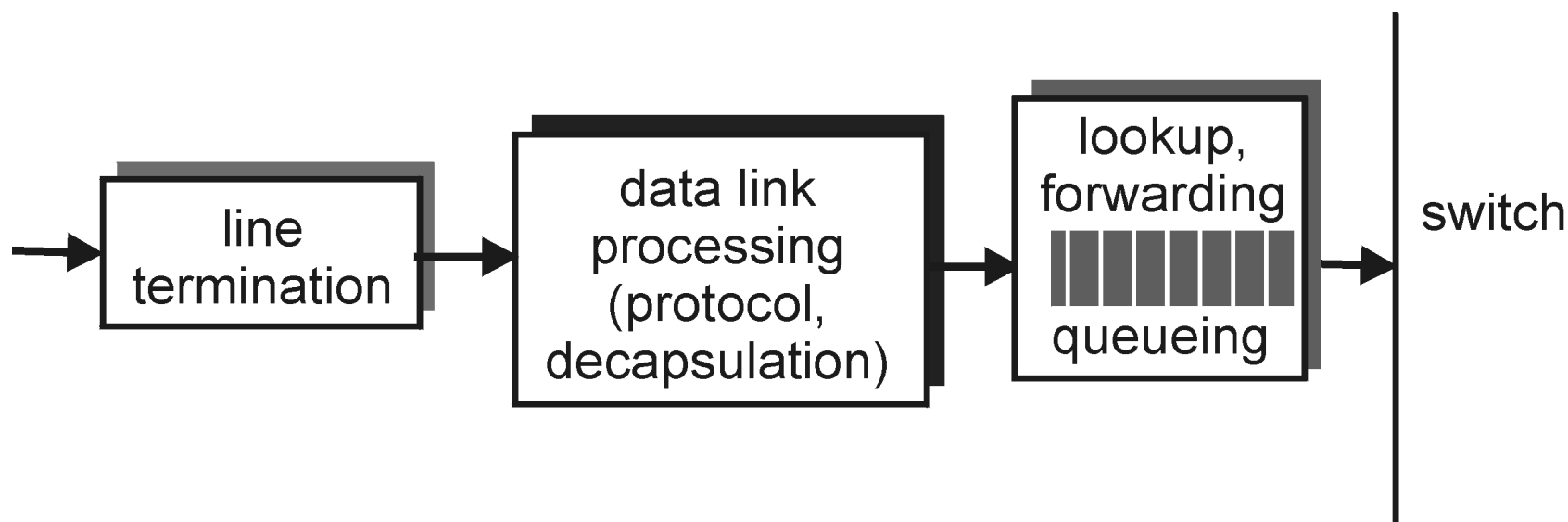
- porta di ingresso
- commutatore
- processore di routing
- porta di uscita



Cenni di architetture di router [2]

Porta di ingresso:

- funzioni del livello 1
 - funzioni del livello 2
 - funzioni del livello 3 → funzioni di ricerca e forwarding della porta di uscita; ottimizzazione della ricerca nella tabella di routing
- } associate a un singolo link di ingresso

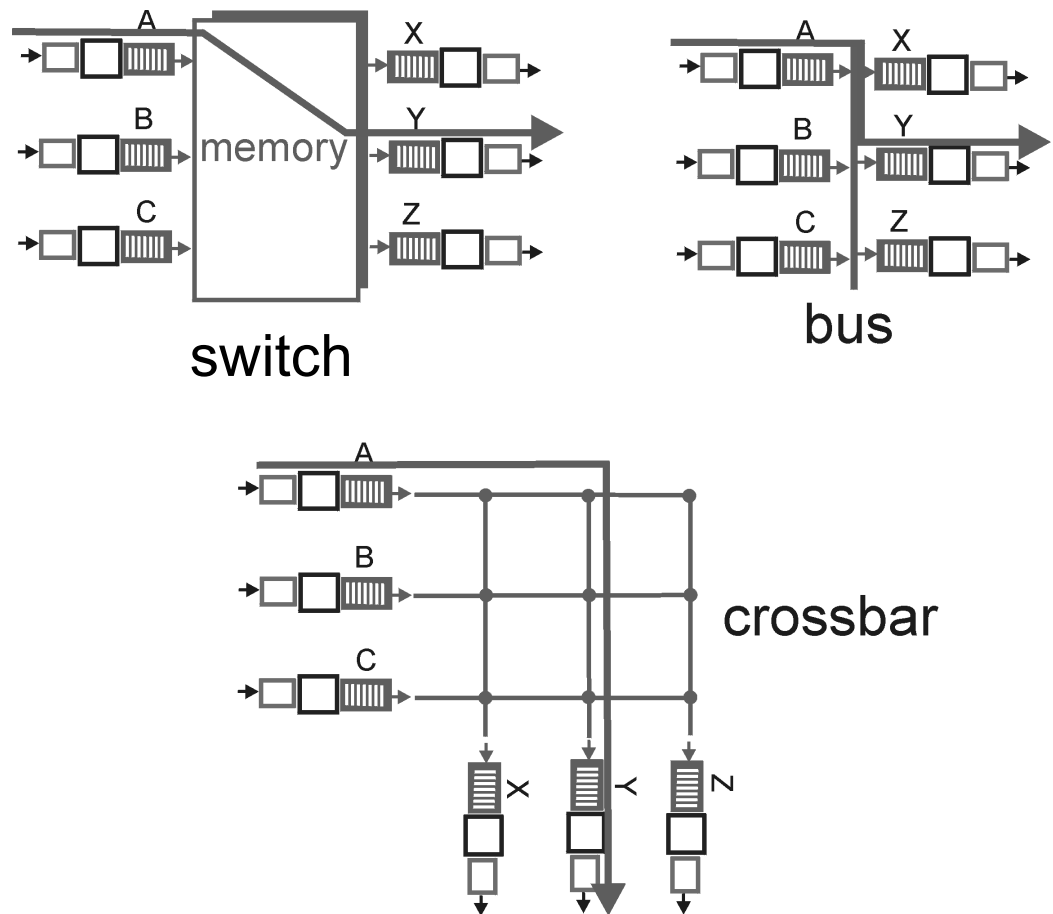


Cenni di architetture di router [3]

Componenti di switching

FUNZIONE: spostamento del pacchetto dalla porta di ingresso a quella di uscita “opportuna”

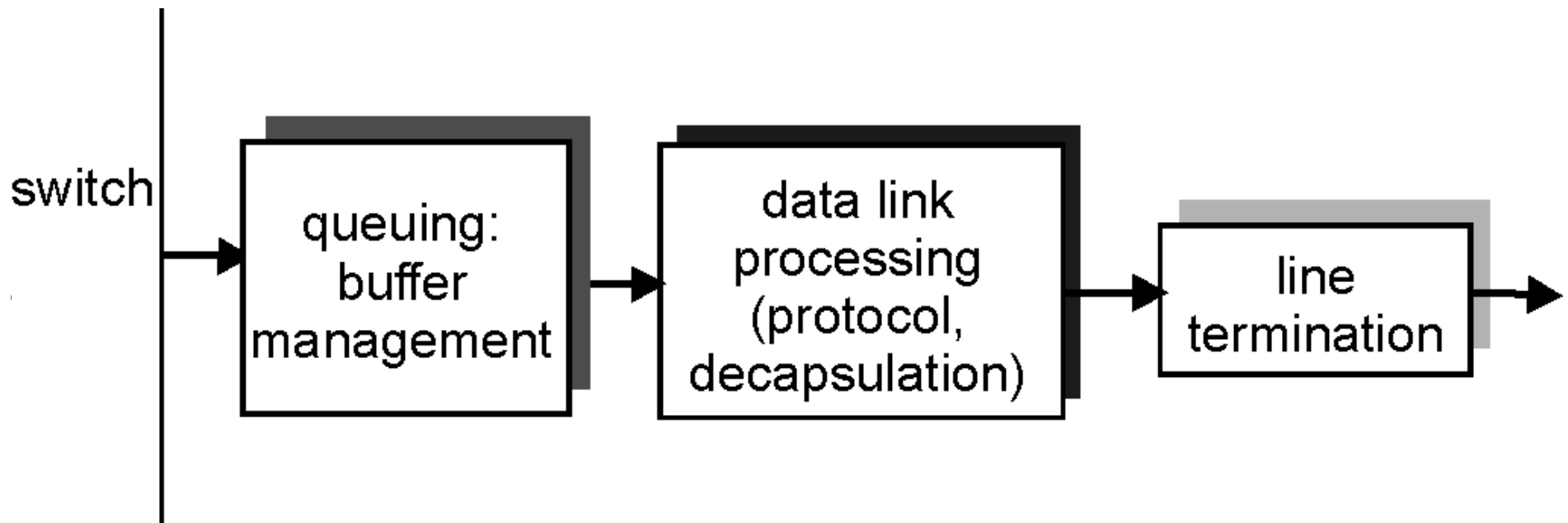
TECNICHE: Commutazione basata su switch, bus o rete di interconnessione crossbar



Cenni di architetture di router [4]

Porta di uscita:

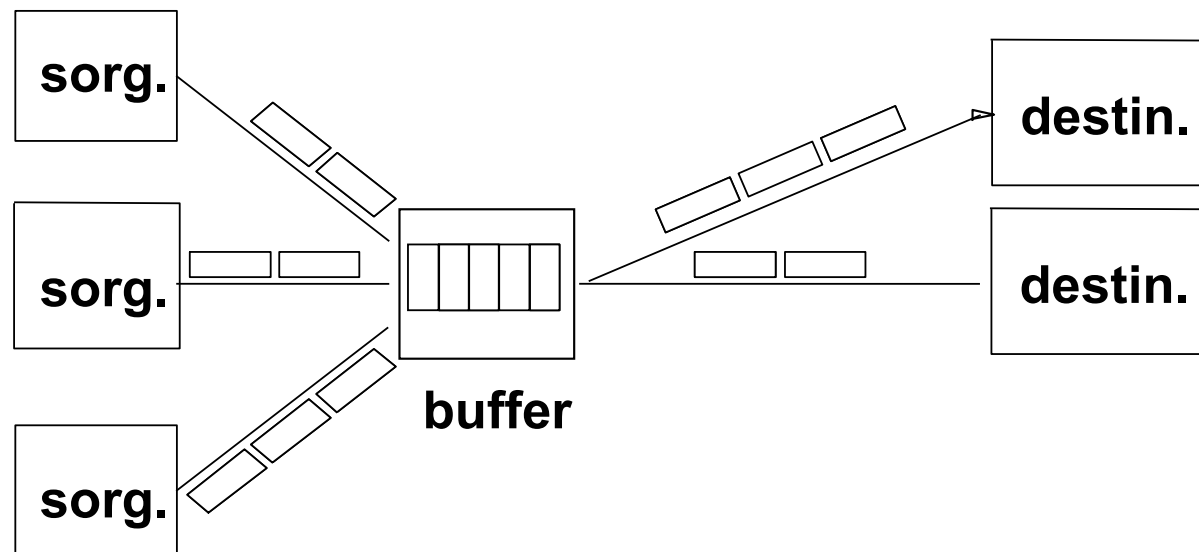
- funzioni del livello 1
 - funzioni del livello 2
 - funzioni del livello 3 → funzioni di gestione della coda e del buffer di uscita (la velocità con cui il commutatore consegna i pacchetti deve essere superiore alla capacità del link di uscita)
- } associate a un singolo link di uscita



Gestione del conflitto

- Si bufferizzano i pacchetti in conflitto per lo stesso link
- Il buffer determina in pratica una coda di pacchetti che può essere processata in ordine **FIFO (First-In-First-Out)**, ma non necessariamente (es., in base alla priorità)

➔ **Congestione = riempimento del buffer**



Trasmissioni e conflitti nel packet switching

Comunicazione store and forward:

(i pacchetti si muovono di un hop alla volta)

1. trasmessi su un link, arrivano ad un router
2. aspettano (presso il router), il loro turno per poter essere trasmessi sul successivo

Conflitto di risorse

- La domanda aggregata di risorse può eccedere la quantità disponibile
- Non essendoci prenotazione, si possono creare **congestioni** (*impreviste*):
 - i pacchetti rimangono accodati (se c'è spazio) in attesa di poter utilizzare il link
 - Se la coda è piena, il pacchetto viene perduto (senza avvisi!)
- Possibilità di utilizzare un link differente a seconda dello stato della rete