



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Internet e IP

Franco CALLEGATI

Dipartimento di Informatica: Scienza e Ingegneria

A.A. 2018-2019



# Nascita di Internet

**Bruce Sterling, autore di fantascienza, scrive sulla rivista "Fiction & Science fiction" nel febbraio '93:**

Circa 30 anni fa, la RAND Corporation, una delle principali fucine di idee degli U.S.A. del periodo della guerra fredda, si trovò ad affrontare un singolare problema strategico. “Come avrebbero potuto comunicare le autorità americane dopo una guerra nucleare?” Gli U.S.A. Post guerra atomica avrebbero comunque avuto bisogno di una rete di comando e controllo, da città a città, da base a base, da stato a stato ... la soluzione proposta dalla RAND fu resa pubblica nel 1964.

In primo luogo la rete *non avrebbe dovuto avere alcuna autorità centrale*. Inoltre doveva essere progettata fin dal principio *per operare anche se a pezzi...*

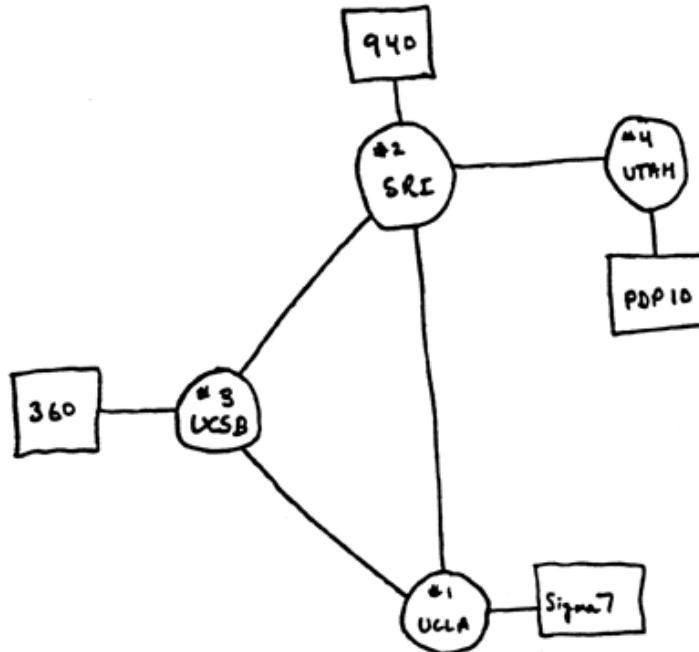


**Il principio era semplice: si doveva assumere che la rete stessa potesse essere inaffidabile in qualunque momento.**



# ARPANet

- 1969: Il dipartimento della difesa USA (DoD) attraverso l' Agenzia per i Progetti di Ricerca Avanzati (ARPA), finanzia la sperimentazione di una rete di calcolatori (ARPANET) fra:
  - UCLA (University of California at Los Angeles)
  - Stanford Research Center
  - UCSB (University of California at Santa Barbara)
  - Università dello Utah





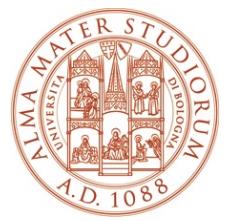
# Il Prof. Leonard Kleinrock e l'IMP



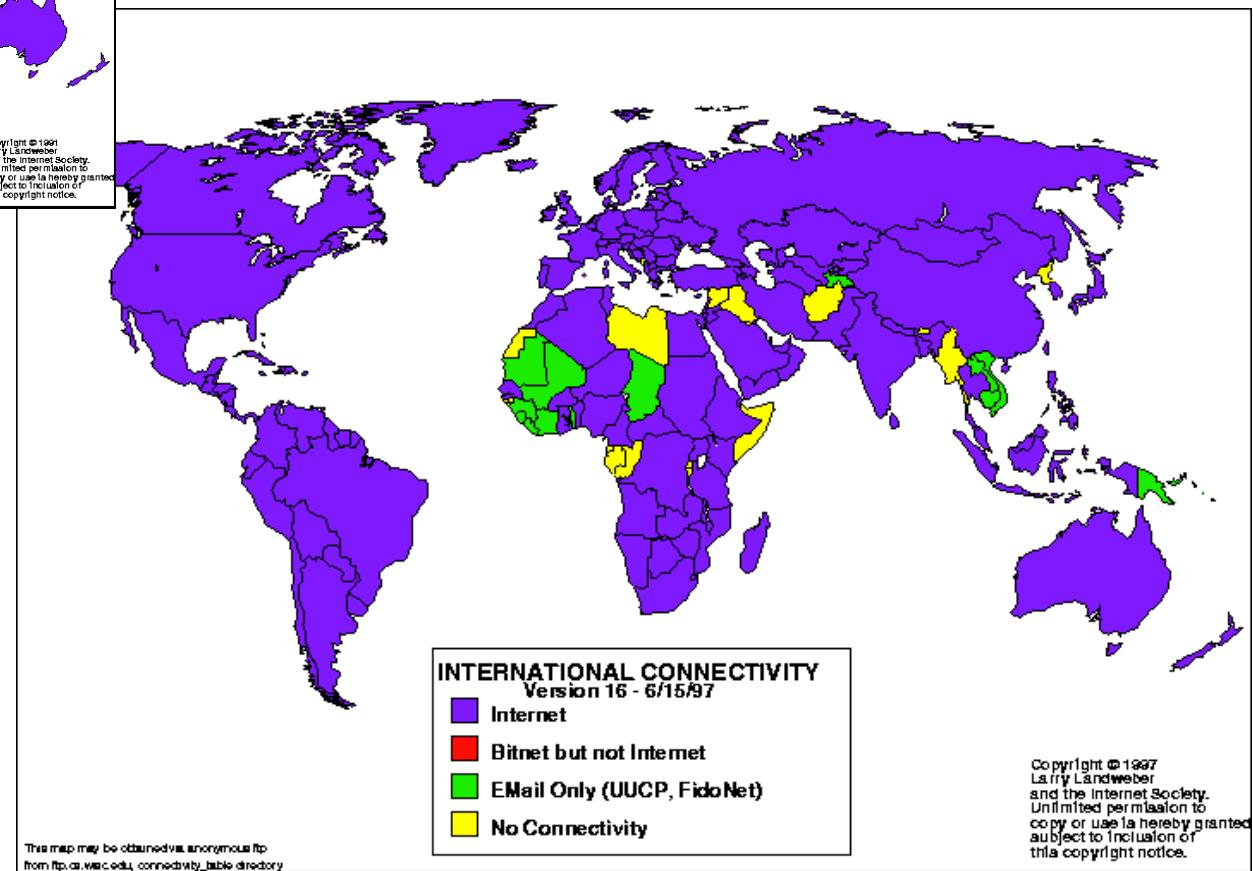
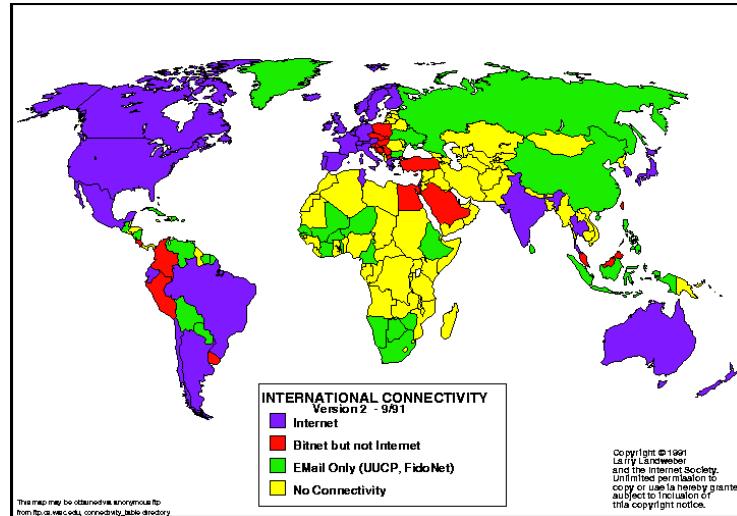


# Un po' di storia...

- 1973: Prima connessione internazionale (Inghilterra e Norvegia).
- 1982: ARPANET adotta come standard i protocolli TCP/IP
  - nasce la prima definizione di Internet come un insieme di reti interconnesse, utilizzanti TCP/IP
  - il 1 Gennaio 1983 ARPANET passa al TCP/IP
  - L'università della California a Berkley rende di pubblico dominio il codice sorgente del TCP/IP
- 1986: Arpanet viene potenziata creando NSFNET (National Science Foundation NETwork).
- 1989: il numero di hosts supera 100.000. L'Italia si connette a NSFNET e quindi a Internet.
- 1990: ARPANET cessa di esistere, ma tutte le sue funzioni e infrastrutture rimangono operanti inalterate.
- 1992: viene rilasciato dal CERN il World Wide Web (WWW) ed il numero di hosts supera 1.000.000.
- 1993: la Casa Bianca e le Nazioni Unite si connettono ad Internet.
- 1994: iniziano i primi servizi commerciali e i primi centri commerciali virtuali. E' possibile ordinare una pizza direttamente tramite Internet presso "Pizza HUT".
- 1995: si vieta il trasporto di traffico di tipo commerciale su NSFNET, che rimane ad uso esclusivo degli enti di ricerca.
- 1996: WWW diventa il primo servizio di Internet in termini di quantità di bytes trasportati.

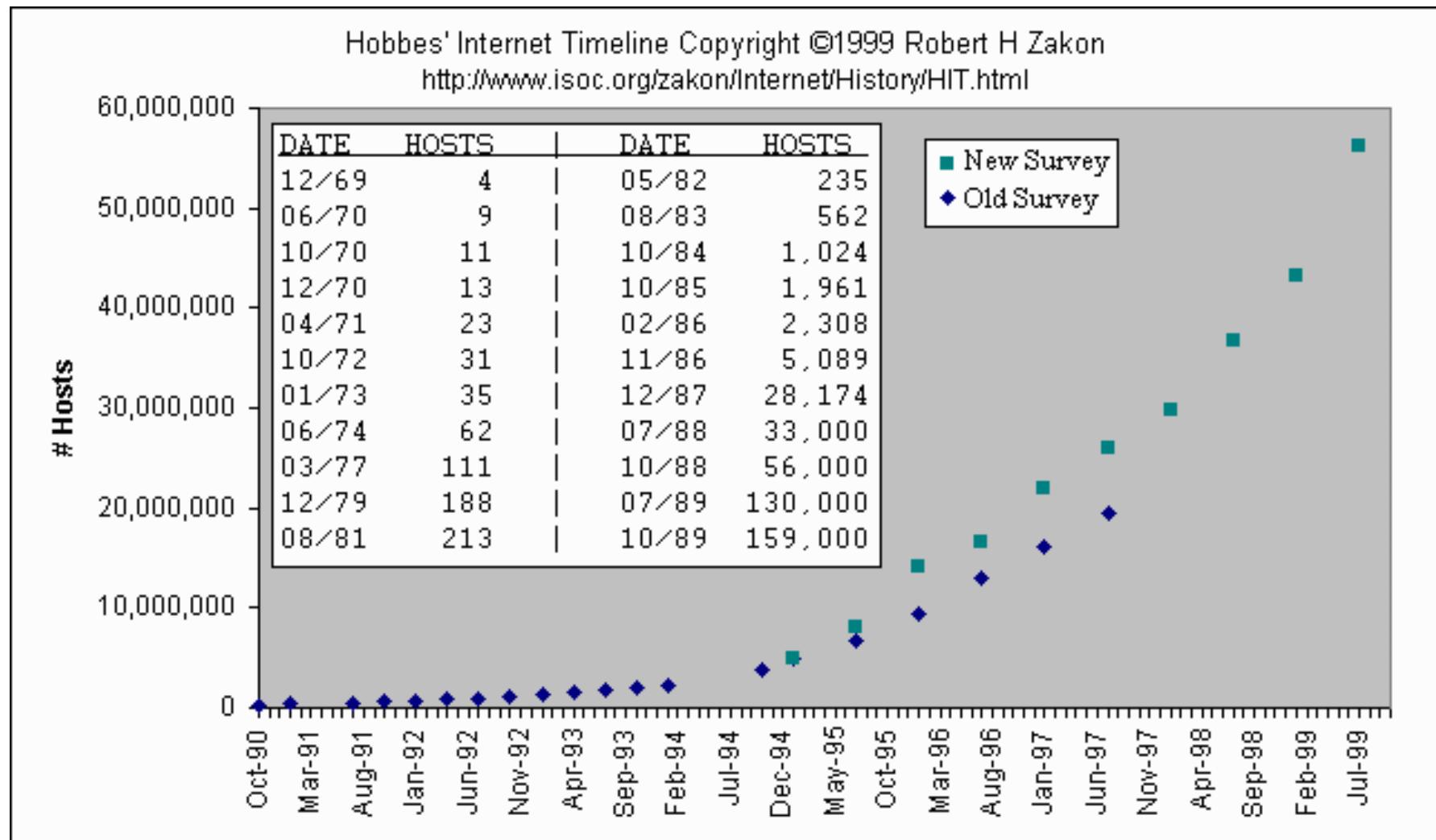


# Diffusione di Internet



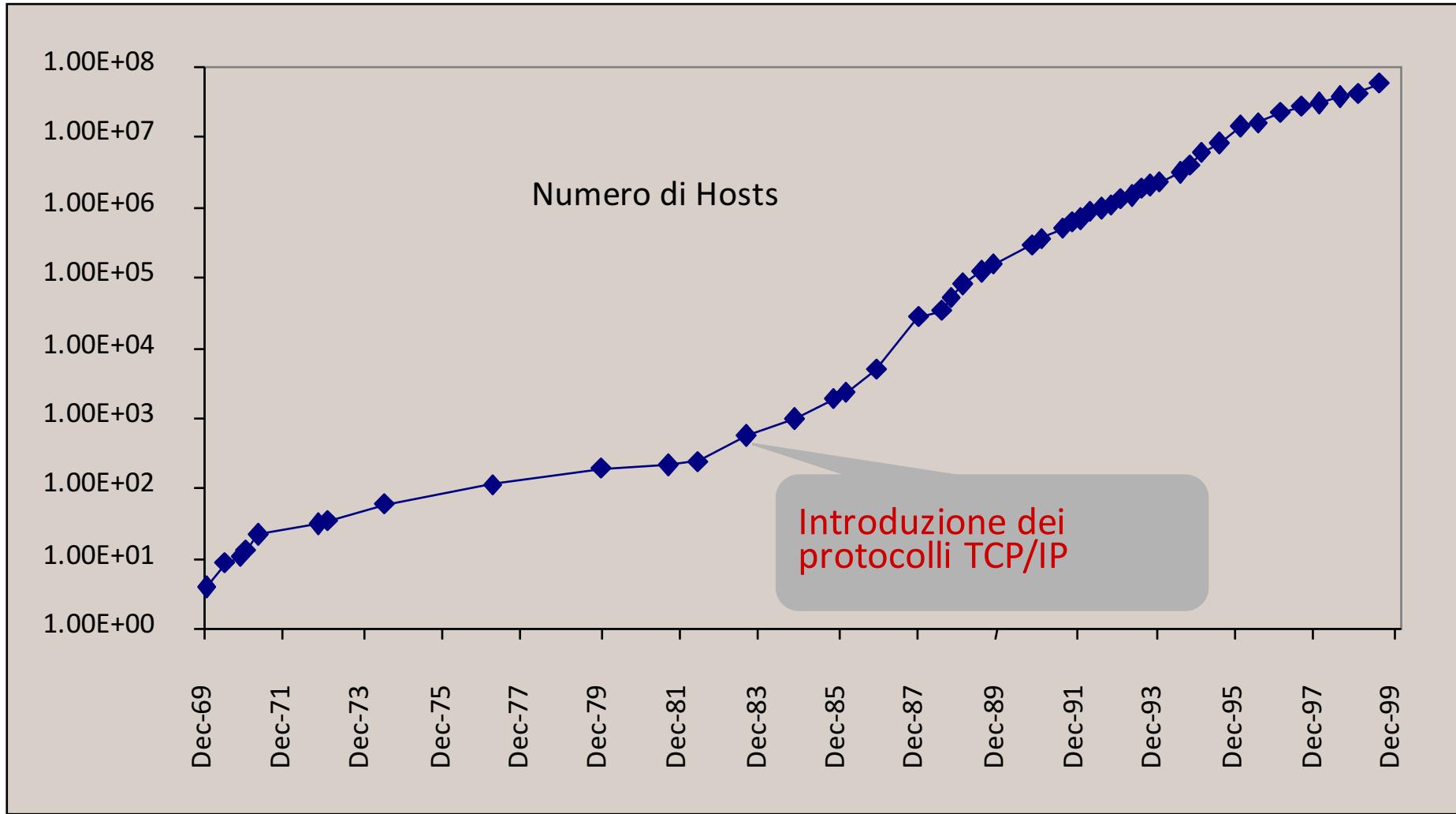


# La crescita di Internet





# La crescita di Internet





# Standard di Internet: Enti di gestione

- Non esistono veri e propri enti che svolgono la funzione di gestione, ma solo enti di coordinamento delle attività di ricerca e di sviluppo che ora convergono nella Internet Society
- Da questa ora dipende il cosiddetto Internet Advisory Board (IAB) e si compone di due sottogruppi principali
  - **Internet Engineering Task Force (IETF)**, con lo scopo di coordinare le attività di ingegnerizzazione ed implementazione
  - **Internet Research Task Force (IRTF)**, con lo scopo di coordinare le attività di ricerca





# RFC

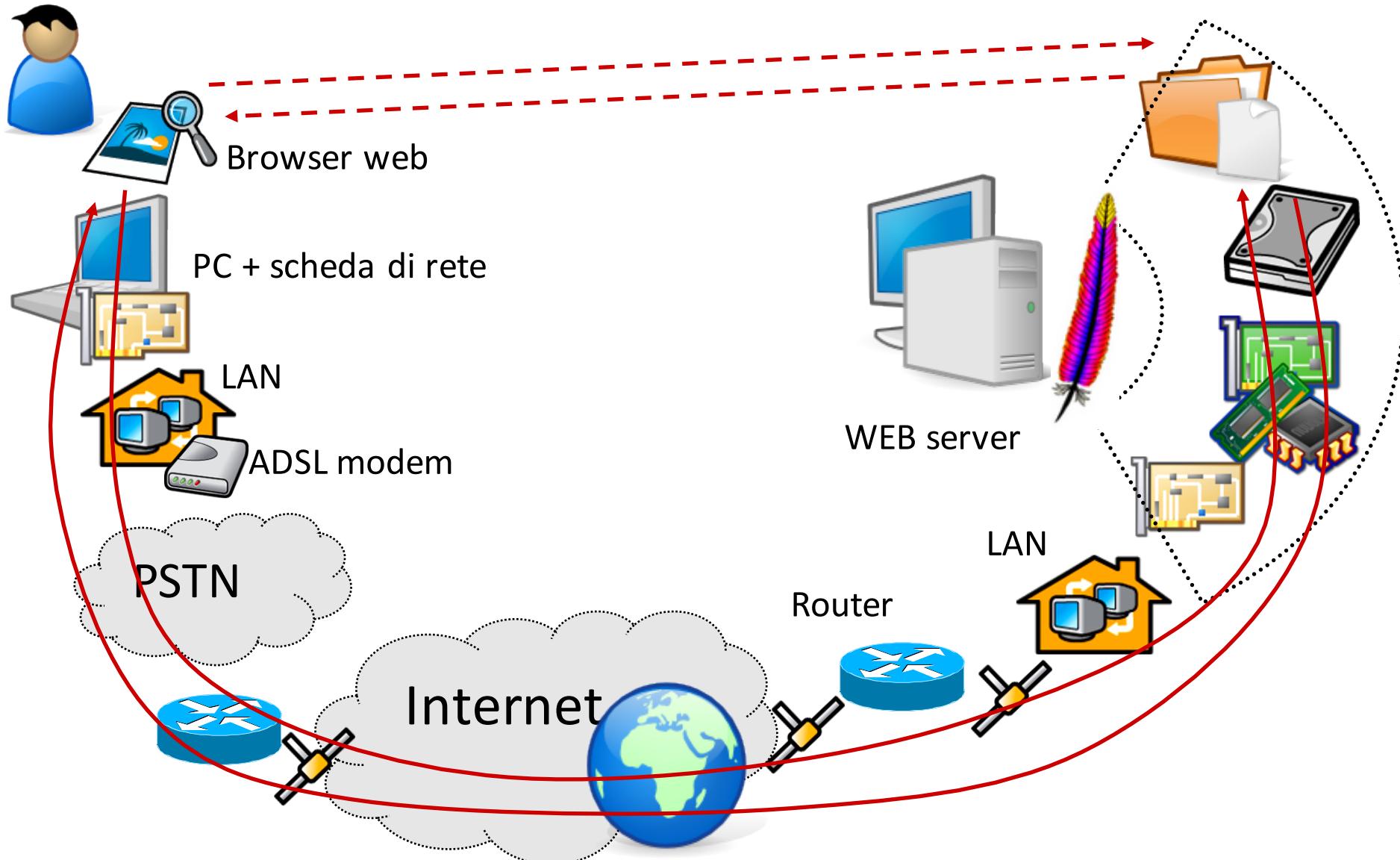
- I protocolli sono frutto del lavoro di gruppi di ricerca
- I vari protocolli sono definiti in documenti detti **Request For Comment (RFC)**
- **RFC** sono documenti di pubblico dominio, distribuiti liberamente a chiunque li richieda, consultabili all' indirizzo  
<http://www.ietf.org/rfc.html>
- Alcuni RFC diventano Internet Standard
  - Prima Proposed Standards
  - Poi Draft standards

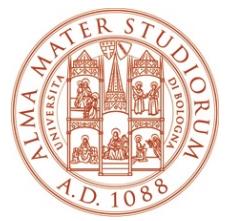


# Altri Enti importanti

- NSF ha fondato un ente chiamato **InterNIC** (**Network Information Center**) allo scopo di fornire alcuni servizi specifici relativi ad Internet:
  - Registrazione di nuove reti e domini
  - Manutenzione di indici e database degli enti interconnessi
  - Servizi di tipo informativo sulla rete
- **IANA = Internet Assigned Number Authority**
  - Mantiene i database dei numeri che hanno significati convenzionali nei protocolli di Internet

# Utilizzo di Internet: chi è coinvolto?





# Indirizzamento

- La comunicazione coinvolge due o più entità
  - Come fa il chiamante a specificare il chiamato?
- Diversi modi di “indirizzare”:
  - Esseri umani
    - Fanno riferimento a nomi simbolici facilmente ricordabili e scrivibili
  - Nodi di commutazione
    - Fanno riferimento a indirizzi, tipicamente numerici ben definiti e standardizzati
  - Sistemi di sicurezza
    - Fanno riferimento alle identità (certificate in qualche modo)
  - Applicazioni
    - Fanno riferimento a identificativi opportunamente definiti



# In Internet

- Tipicamente dobbiamo distinguere fra
  - Identifier
    - Identificativo di una certa risorsa di rete
  - Locator
    - Indirizzo necessario per localizzare tale risorsa
- Vengono definiti
  - Uniform Resource Identifier o URI
  - Uniform resource locator o URL



# Alcuni esempi

- Mobilità
  - Un terminale si sposta da una rete all'altra
    - Locator cambia nel tempo
- Multi-homing
  - Un terminale è connesso con più interfacce a infrastrutture diverse
    - Molteplici locator attivi in contemporanea
    - Potenzialmente un unico identifier



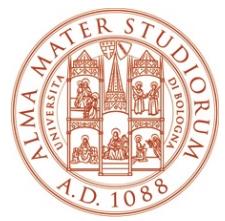
# Indirizzo globale e indirizzo locale

- **Indirizzo *globale***

- È valido per tutta la rete
- Deve essere univoco (non devono esistere indirizzi replicati) per evitare ambiguità
- Va “assegnato” seguendo una procedura di gestione “globale” che assicura la non replicazione

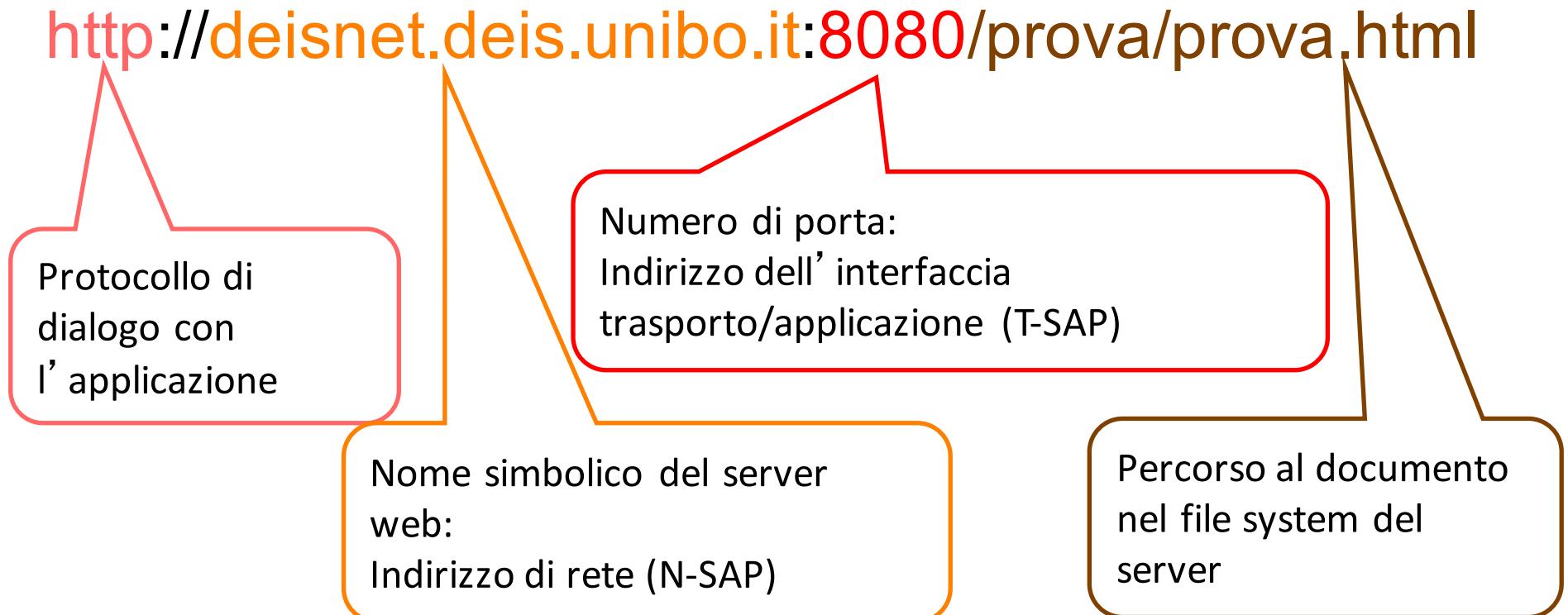
- **Indirizzi *locale***

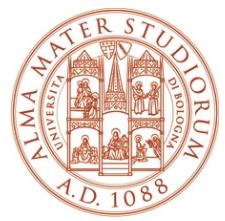
- È valido limitatamente ad un certa sottoporzione della rete
  - Internamente ad un terminale
  - In un dominio di rete specifico
- Può non essere univoco
- Può essere assegnato con una procedura puramente “locale”



# Indirizzamento nel web

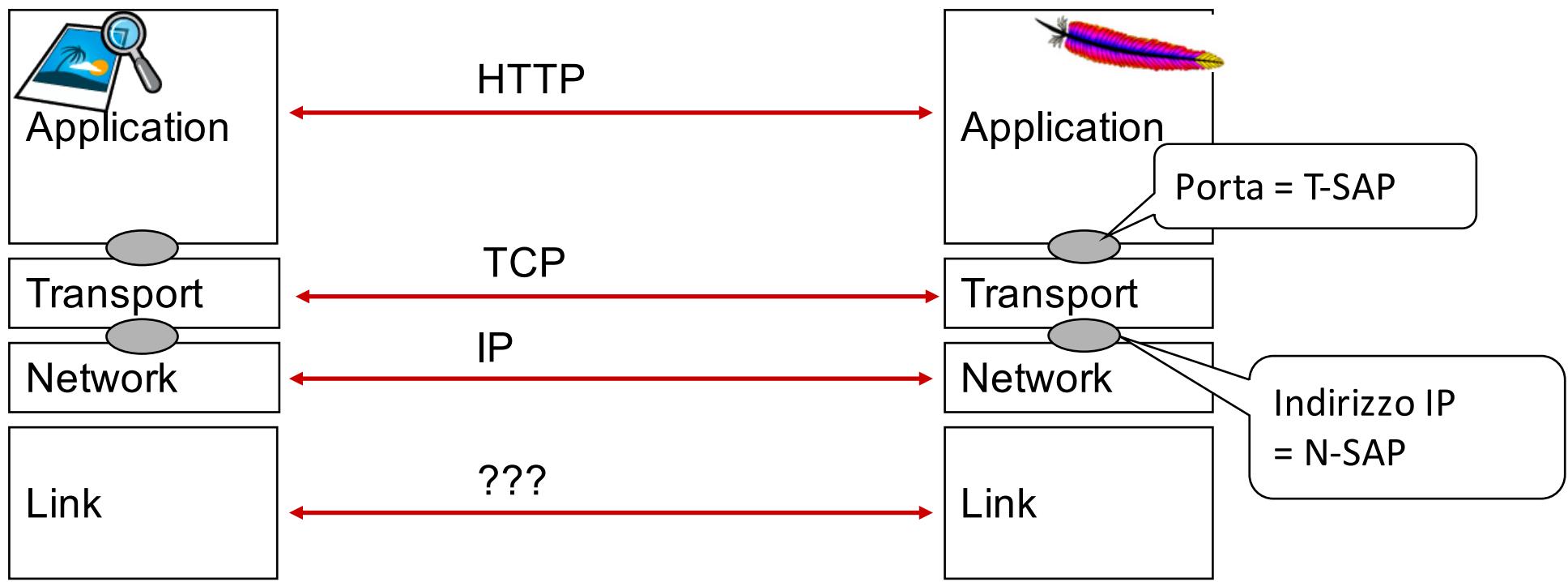
- La risorsa R è univocamente identificata da un indirizzo che la localizza (locator)
  - Uniform Resource Locator (URL)
    - URL è un indirizzo complesso che riflette l'organizzazione a livelli della rete





# Protocolli ed interfacce

- Le applicazioni sono locali al calcolatore (terminale)
  - Alcune parti dell'URL hanno validità locale
    - Numero di porta
    - Percorso nel filesystem
- Il calcolatore va identificato univocamente su Internet
  - Almeno una parte dell'indirizzo deve avere significato unico e universale
    - Indirizzo di rete (numero IP)





# Analisi di protocollo

- Esistono strumenti software per analizzare il traffico di rete
  - Wireshark <http://www.wireshark.org/>

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: tcp.stream eq 0 Expression... Clear Apply

| No. | Time     | Source         | Destination    | Protocol | Info  |
|-----|----------|----------------|----------------|----------|---|
| 1   | 0.000000 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1           |
| 2   | 0.001129 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 3   | 0.001155 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=1 Ack=1 Win=65535 Len=0                          |
| 4   | 0.001372 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET / HTTP/1.1  |
| 5   | 0.002667 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [ACK] Seq=1 Ack=355 Win=6432 Len=0                         |
| 6   | 0.006827 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 7   | 0.008078 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 8   | 0.008127 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=355 Ack=2921 Win=65535 Len=0                     |
| 9   | 0.011568 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 10  | 0.012807 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 11  | 0.012863 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=355 Ack=5841 Win=65535 Len=0                     |
| 12  | 0.014071 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 13  | 0.014109 | 137.204.57.85  | 192.168.10.199 | HTTP     | HTTP/1.1 200 OK (text/html)   |
| 14  | 0.014139 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=355 Ack=7433 Win=65535 Len=0                     |
| 15  | 0.036538 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET /Stili/STYLE_DEISNet.css HTTP/1.1                                 |
| 16  | 0.041114 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 17  | 0.042346 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |

Frame 6: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)  
Ethernet II, Src: DellComp\_ec:46:62 (00:b0:d0:ec:46:62), Dst: DellComp\_89:b3:e9 (00:06:b5:89:b3:e9)  
Internet Protocol, Src: 137.204.57.85 (137.204.57.85), Dst: 192.168.10.199 (192.168.10.199)  
Transmission Control Protocol, Src Port: http (80), Dst Port: ddt (1052), Seq: 1, Ack: 355, Len: 1460

0000 00 06 5b 89 b3 e9 00 b0 d0 ec 46 62 08 00 45 00 ..[....].. Fb..E.  
0010 05 dc 35 70 40 00 3f 06 72 1b 89 cc 39 55 c0 a8 ..5p@?. r...9U..  
0020 0a c7 00 50 04 1c 14 3c 5d 6e 35 37 50 72 50 10 ...P...< Jn57PrP.  
0030 19 20 85 c0 00 00 48 54 54 50 2f 31 2e 31 20 32 ..HT.TS  
0040 30 30 20 4f 4b 0d 0a 44 61 74 65 3a 20 54 75 65 00 0K..  
0050 2c 20 31 33 20 4a 61 6e 20 32 30 30 34 20 31 36 , 13 Jan 2004 16  
0060 3a 31 36 3a 30 31 20 47 4d 54 0d 0a 53 65 72 76 :16:01 G MT..Serv  
0070 65 72 3a 20 41 70 61 63 68 65 2f 32 2e 30 2e 34 er: Adac he/2.0.4

File: "/Users/franco/Docume... | Packets: 31 Displayed: 31 Marked: 0 Load Default

Funzioni di controllo del software

Qui viene mostrata la sequenza dei messaggi  
Una riga per messaggio

Contenuto del messaggio evidenziato sopra  
Vengono evidenziati i vari protocolli utilizzati nel messaggio

Contenuto del messaggio in forma di sequenza di byte



# Esempi di URL

- Interrogazione del server `deisnet.deis.unibo.it` utilizzando porte e percorsi ai documenti standard e non standard
- 1 - porta 80 (default), file `index.php` (default)

<http://deisnet.deis.unibo.it/Didattica/CorsiCE/RetiLA/testpage/>

- 2 - porta 80 (default), file non di default

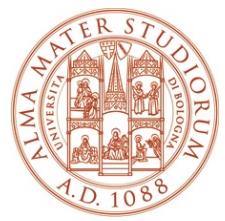
<http://deisnet.deis.unibo.it/Didattica/CorsiCE/RetiLA/testpage/testpage.php>

- 3 - porta 12345 (non default), file di default, URL diverso

<http://deisnet.deis.unibo.it:12345/testpage/>

- 4 - porta 12345 (non default), file non di default

<http://deisnet.deis.unibo.it:12345/testpage/testpage.php>



# Protocollo di trasporto

- Il protocollo di trasporto si occupa del trasporto dei dati end-to-end
- Può trasportare i dati pertinenti ad una qualunque applicazione
- I flussi dati di diverse applicazioni sono distinguibili sulla base del numero di porta
- Esistono diversi protocolli di trasporto
  - UDP – User Datagram Protocol
  - TCP – Transmission Control Protocol
  - RTP – Real Time Transmission Protocol
    - Il protocollo di trasporto per una determinata istanza applicativa viene scelto in funzione delle caratteristiche che il trasporto dei dati deve avere



# Numero di porta

- Indirizzo di 16 bit
  - Valori decimali da 0 a 65535
- Locale al singolo calcolatore, ripetute su tutti i calcolatori
- Condiviso fra tutti i protocolli di trasporto

Screenshot of Wireshark network traffic analysis tool showing a TCP session between two hosts.

The packet list shows:

| No. | Time     | Source         | Destination    | Protocol | Info  |
|-----|----------|----------------|----------------|----------|---|
| 1   | 0.000000 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1           |
| 2   | 0.001129 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 3   | 0.001155 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=1 Ack=1 Win=65535 Len=0                          |
| 4   | 0.001372 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET / HTTP/1.1  |
| 5   | 0.002667 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [ACK] Seq=1 Ack=355 Win=6432 Len=0                         |
| 6   | 0.006827 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 7   | 0.008078 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |

The details pane shows the selected packet (Transmission Control Protocol) with the following fields:

- Source port: ddt (1052)
- Destination port: http (80)
- [Stream index: 0]
- Sequence number: 0 (relative sequence number)
- Header length: 28 bytes
- Flags: 0x02 (SYN)
- Window size: 65535
- Checksum: 0x6bdc [validation disabled]
- Options: (8 bytes)

The bytes pane shows the raw hex and ASCII data of the selected packet:

|      |   |                   |
|------|---|-------------------|
| 0000 | 00 b0 d0 ec 46 62 00 06 5b 89 b3 e9 08 00 45 00 | ....Fb.. [.....E. |
| 0010 | 00 30 01 4b 40 00 80 06 00 00 c0 a8 0a c7 89 cc | .0.K@.....        |
| 0020 | 39 55 04 1c 00 50 35 37 4f 0f 00 00 00 00 70 02 | 9U...P57 0.....p. |
| 0030 | ff ff 6b dc 00 00 02 04 05 b4 01 01 04 02       | ..k.....          |

At the bottom, status bar: Frame (frame), 62 bytes | Packets: 31 Displayed: 31 Marked: 0 Load time: 0:00.005 | Profile: Default

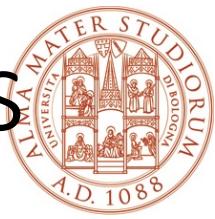


# Classificazione

- Regole d' uso
  - Da 1 a 1023 (in origine da 1 a 255): **Riservati**
    - possono essere usati solo dai server
  - da 1024 a 49151: **Registrati**
    - Sono usati da alcuni servizi ma anche da client
  - Da 49151 a 65535: **ad uso dei client**
- Una parte dei numeri di porta sono riservati (Well Known Ports)

| #   | Protocol                  | Servizio  |
|-----|---------------------------|---|
| 21  | FTP-CONTROL               | File Transfer Protocol                                      |
| 20  | FTP-DATA                  | File Transfer Protocol                                      |
| 23  | TELNET                    | Accesso via terminale                                       |
| 25  | SMTP                      | Trasferimento di posta elettronica                          |
| 53  | DNS                       | Accesso al DNS  |
| 80  | HTTP                      | Web server  |
| 109 | POP2                      | Lettura posta elettronica                                   |
| 22  | SSH                       | Accesso via terminale cifrato                               |
| 110 | POP3                      | Lettura posta elettronica                                   |
| 137 | NETBIOS Name Service.     | Servizio di rete per applicazioni in ambiente DOS (Windows) |
| 138 | NETBIOS Datagram Service. |   |
| 139 | NETBIOS Session Service.  |   |
| 443 | HTTPS                     | WEB cifrato   |

# IANA (Internet Assigned Numbers Authority)



## PORTE NUMBERS

(last updated 2010-09-21)

The port numbers are divided into three ranges: the Well Known Ports, the Registered Ports, and the Dynamic and/or Private Ports.

The Well Known Ports are those from 0 through 1023.

DCCP Well Known ports SHOULD NOT be used without IANA registration. The registration procedure is defined in [RFC4340], Section 19.9.

The Registered Ports are those from 1024 through 49151

DCCP Registered ports SHOULD NOT be used without IANA registration. The registration procedure is defined in [RFC4340], Section 19.9.

The Dynamic and/or Private Ports are those from 49152 through 65535



# Protocollo di rete

- Garantisce il corretto indirizzamento ed instradamento dei dati
- Deve necessariamente essere unico in una rete globale
- Internetworking Protocol - IP

Screenshot of Wireshark network traffic analysis tool showing a session between two hosts. A red box highlights the Internet Protocol details for the second packet, and an orange callout box points to the source and destination IP addresses.

Filter: tcp.stream eq 0

| No. | Time     | Source         | Destination    | Protocol | Info  |
|-----|----------|----------------|----------------|----------|---|
| 1   | 0.000000 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1           |
| 2   | 0.001129 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 3   | 0.001155 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=1 Ack=1 Win=65535 Len=0                          |
| 4   | 0.001372 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET / HTTP/1.1  |
| 5   | 0.002667 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [ACK] Seq=1 Ack=355 Win=6432 Len=0                         |
| 6   | 0.006827 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 7   | 0.008078 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |

Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)  
Internet Protocol, Src: 192.168.10.199 (192.168.10.199), Dst: 137.204.57.85 (137.204.57.85)

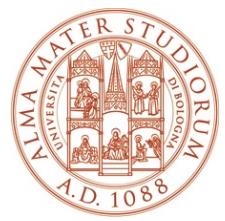
Version: 4  
Header length: 20 bytes  
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)  
Total Length: 48  
Identification: 0x014b (331)  
Flags: 0x02 (Don't Fragment)  
Fragment offset: 0  
Time to live: 128  
Protocol: TCP (6)  
Header checksum: 0x0000 [incorrect, should be 0x6aec]  
Source: 192.168.10.199 (192.168.10.199)  
Destination: 137.204.57.85 (137.204.57.85)

Transmission control protocol, Src Port: ddt (1024), Dst Port: http (80), Seq: 0, Len: 0

0000 00 b0 d0 ec 46 62 00 06 5b 89 b3 e9 08 00 45 00 ...Fb.. [.....E.  
0010 00 30 01 4b 40 00 80 06 00 00 c0 a8 0a c7 89 cc .0.K@...  
0020 39 55 04 1c 00 50 35 37 4f 0f 00 00 00 70 02 9U...P57 0.....p.  
0030 ff ff 6b dc 00 00 02 04 05 b4 01 01 04 02 ..k.....

File: "/Users/franco/Docume... Packets: 31 Displayed: 31 Marked: 0 Load time: 0:00.005 Profile: Default

Indirizzi IP del chiamato  
(sorgente) e del chiamante  
(destinazione)



# Protocollo di rete

- Garantisce il corretto indirizzamento ed instradamento dei dati
- Deve necessariamente essere unico in una rete globale
- Internetworking Protocol - IP

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: tcp.stream eq 0 Expression... Clear Apply

| No. | Time     | Source         | Destination    | Protocol | Info  |
|-----|----------|----------------|----------------|----------|---|
| 1   | 0.000000 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1           |
| 2   | 0.001129 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 3   | 0.001155 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=1 Ack=1 Win=65535 Len=0                          |
| 4   | 0.001372 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET / HTTP/1.1  |
| 5   | 0.002667 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [ACK] Seq=1 Ack=355 Win=6432 Len=0                         |
| 6   | 0.006827 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 7   | 0.008078 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |

► Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)  
► Ethernet II, Src: DellComp\_89:b3:e9 (00:06:5b:89:b3:e9), Dst: DellComp\_ec:46:62 (00:b0:d0:ec:46:62)  
▼ Internet Protocol, Src: 192.168.10.199 (192.168.10.199), Dst: 137.204.57.85 (137.204.57.85)  
    Version: 4  
    Header length: 20 bytes  
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)  
    Total Length: 48  
    Identification: 0x014b (331)  
    Flags: 0x02 (Don't Fragment)  
    Fragment offset: 0  
    Time to live: 128  
    Protocol: TCP (6)  
    ► Header checksum: 0x0000 [incorrect, should be 0x6aec]  
        Source: 192.168.10.199 (192.168.10.199)  
        Destination: 137.204.57.85 (137.204.57.85)  
    ► Transmission Control Protocol, Src Port: ddt (1052), Dst Port: http (80), Seq: 0, Len: 0

0000 00 b0 d0 ec 46 62 00 06 5b 89 b3 e9 08 00 45 00 ....Fb. ....E.  
0010 00 30 01 4b 40 00 80 06 00 00 c0 a8 0a c7 89 cc .0.K@....  
0020 39 55 04 1c 00 50 35 37 4f 0f 00 00 00 00 70 02 9U...P57 0.....p.  
0030 ff ff 6b dc 00 00 02 04 05 b4 01 01 04 02 ..k.....

File: "/Users/franco/Docume... Packets: 31 Displayed: 31 Marked: 0 Load time: 0:00.005 Profile: Default

Indirizzi IP del chiamato  
(sorgente) e del chiamante  
(destinazione)



# L'indirizzo IP

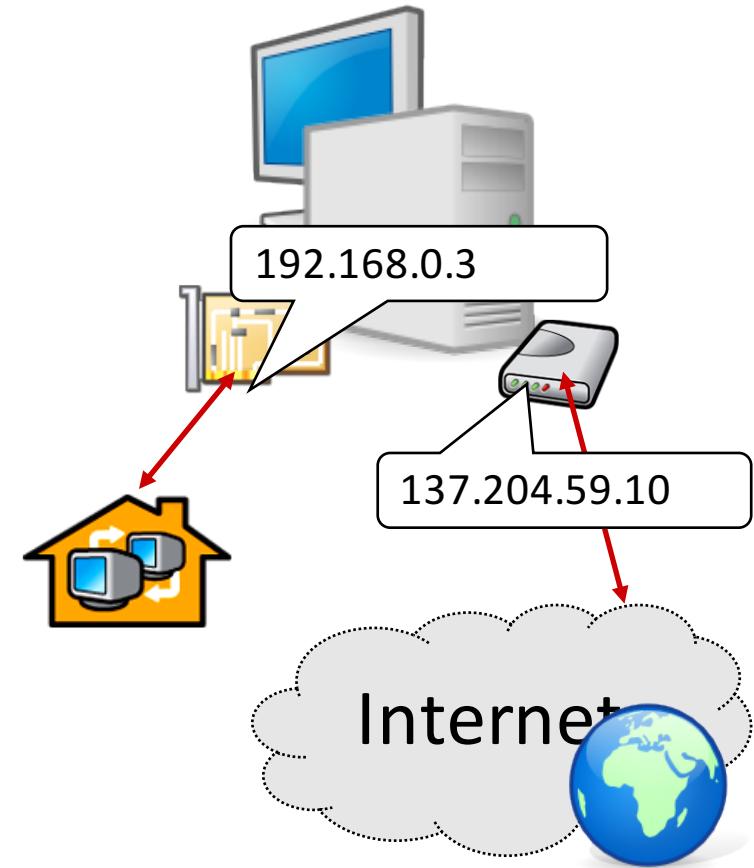
- Indirizzi di lunghezza fissa pari a **32 bit**
- Scritti convenzionalmente come sequenza di 4 numeri decimali, con valori da **0** a **255**, separati da punto (rappresentazione **dotted decimal**)

10001001.11001100.11010100.00000001  
137.204.212.1

- Numero teorico max. di indirizzi  
 **$2^{32} = 4.294.967.296$** 
  - In realtà si riesce a sfruttare un numero molto inferiore

# Indirizzi e interfacce di rete

- L' indirizzo identifica i punti di interconnessione di un host con la rete
  - Non identifica un host individuale, ma una delle sue interfacce di rete
- **Multi-homed hosts**
  - host con due o più interfacce di rete
- Esempio: un router che collega  $N$  reti ha
  - $N$  interfacce di rete
  - $N$  distinti indirizzi IP, uno per ogni interfaccia di rete





# Infrastruttura fisica di accesso alla rete

- Tipicamente un calcolatore si connette alla rete tramite una rete di accesso locale detta LAN
- LAN = Local Area Network
- Le tecnologie LAN tipicamente implementano strato 2 e strato 1 secondo il modello ISO-OSI
  - Adottando soluzioni adatta al collegamento su breve distanza e finalizzate a
    - Canale di elevata capacità (bit rate > 10-100 Mbit/s)
    - Costi limitati (accessibili all'utente finale)
- Le tecnologie LAN oggi più comuni sono state sviluppate adottando un canale di trasmissione/ricezione condiviso fra tutti i calcolatori della LAN



# Canale condiviso

- **Implicazioni**
  - Comunicazione broadcast: uno parla tutti sentono
- **Problemi**
  - Controllo dell'accesso al canale
    - Come evitare di parlare uno sull'altro rendendo la comunicazione incomprensibile
  - Limitazione della quantità di dati da elaborare per lo strato 3
    - Se tutti i dati vengono ricevuti lo strato 3 dovrebbe elaborare anche le conversazioni degli altri
- **È necessario un indirizzo LAN**
  - L'interfaccia di strato 2 (che parla e riceve con la LAN) legge e passa allo strato 3 solamente i dati di sua pertinenza
- **Chi decide gli indirizzi per la LAN?**

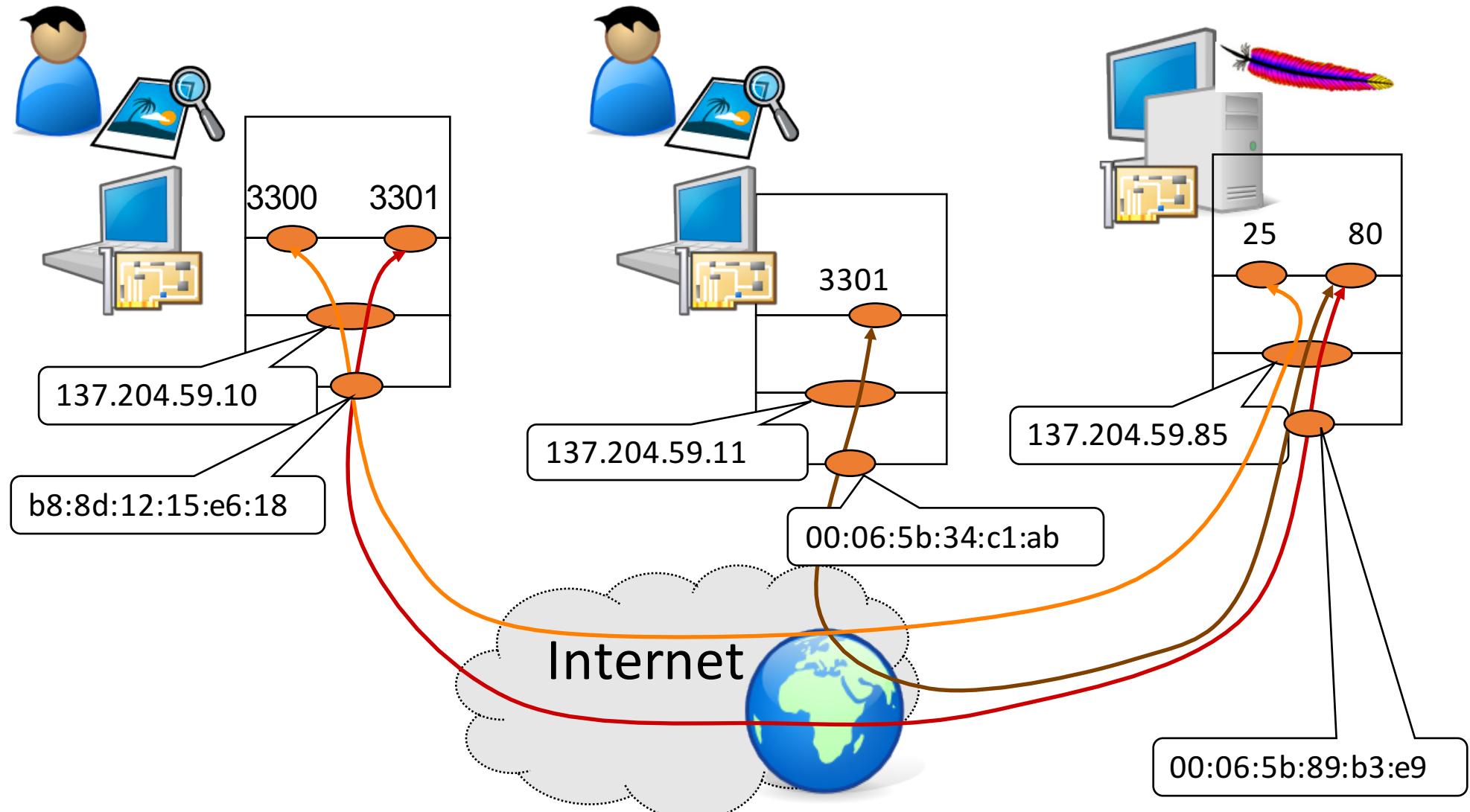


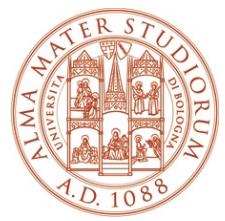
# Gli indirizzi LAN (MAC address)

- Sono composti da 48 bit (6 byte)
- Sono cablati nella scheda di rete
- Sono univoci a livello mondiale
  - i primi 3 byte individuano il costruttore
    - <http://www.macvendorlookup.com>
  - I secondi 3 numerano progressivamente le schede
- E' possibile specificare
  - un singolo destinatario (unicast)
    - 00-60-b0-78-e8-fd
  - un indirizzo di gruppo (multicast)
    - il primo bit deve essere a 1
  - un invio a tutte le stazioni (broadcast)
    - ff-ff-ff-ff-ff-ff



# Flussi di comunicazione





# Connessioni

- Per identificare il singolo flusso è sufficiente conoscere:
  - IP sorgente
  - IP destinazione
  - Porta sorgente
  - Porta destinazione
- Nell'esempio precedente sono attivi
  - 137.204.59.10:3300  $\longleftrightarrow$  137.204.57.85:25
  - 137.204.59.10:3301  $\longleftrightarrow$  137.204.57.85:80
  - 137.204.59.11:3301  $\longleftrightarrow$  137.204.57.85:80



# Implementazioni dei servizi in Internet

- Comunicazioni fra calcolatori (Host) = scambio di messaggi fra processi applicativi (Applicazioni)
  - Un messaggio in arrivo ad un host è utilizzabile se è in esecuzione (running) un processo applicativo che legge il messaggio e sa cosa farsene
- **Client-server**
  - Nel modello classico gli host in rete sono classificabili in due tipologie:
    - *Server*: mettono a disposizione risorse di elaborazione e dati
    - *Client*: ospitano applicazioni che, al fine di svolgere le relative funzioni, si connettono ai server per ottenere risorse ed informazioni
  - La variante **Peer-to-peer** (P2P)
    - Gli host in rete sono tutti equivalenti (peer, appunto) e fungono alternativamente sia da client che da server verso altri nodi
      - In una rete P2P qualsiasi nodo utilizza e mette a disposizione contemporaneamente risorse ed informazioni in rete



# Client-server

- Il processo Server si predisponde a ricevere una connessione eseguendo una **apertura passiva**
  - Crea una socket e si mette in ascolto in attesa dell' arrivo di una richiesta di connessione (questo processo nel mondo Unix è chiamato Demone)
- Il processo Client esegue una **apertura attiva** tentando di collegarsi al processo server di destinazione



# La ricerca della destinazione

- Il client deve conoscere indirizzo IP e il Numero di porta del server di destinazione
  - Come fa a scoprirli?
- Nell' URL sono specificati
  - Protocollo applicativo
    - A cui corrisponde una well known port
  - Eventuale numero di porta non standard
  - Il numero IP o il nome del server
    - Il nome deve tramutarsi in un numero IP
    - Il numero IP identifica in modo univoco il punto di accesso alla rete del server
- Come fa il nome a diventare un numero?



# Esempio

Screenshot of Wireshark showing a network capture. The filter is set to `tcp.stream eq 0`. The packet list shows the following sequence:

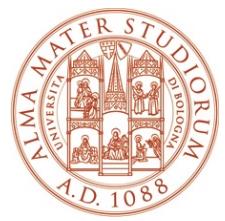
| No. | Time     | Source         | Destination    | Protocol | Info  |
|-----|----------|----------------|----------------|----------|---|
| 1   | 0.000000 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1           |
| 2   | 0.001129 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 3   | 0.001155 | 192.168.10.199 | 137.204.57.85  | TCP      | ddt > http [ACK] Seq=1 Ack=1 Win=65535 Len=0                          |
| 4   | 0.001372 | 192.168.10.199 | 137.204.57.85  | HTTP     | GET / HTTP/1.1  |
| 5   | 0.002667 | 137.204.57.85  | 192.168.10.199 | TCP      | http > ddt [ACK] Seq=1 Ack=355 Win=6432 Len=0                         |
| 6   | 0.006827 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |
| 7   | 0.008078 | 137.204.57.85  | 192.168.10.199 | TCP      | [TCP segment of a reassembled PDU]                                    |

The details pane shows the first TCP segment (Frame 1) with the following fields:

- Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
- Ethernet II, Src: DellComp\_89:b3:e9 (00:06:5b:89:b3:e9), Dst: DellComp\_ec:46:62 (00:b0:d0:ec:46:62)
- Internet Protocol, Src: 192.168.10.199 (192.168.10.199), Dst: 137.204.57.85 (137.204.57.85)
- Transmission Control Protocol, Src Port: ddt (1052), Dst Port: http (80), Seq: 0, Len: 0

Annotations highlight the source port (1052) and destination port (80). A callout box labeled "Well known port del server HTTP" points to the destination port. Another callout box labeled "Numero IP del server" points to the source IP address (192.168.10.199).

Frame (frame), 62 bytes | Packets: 31 Displayed: 31 Marked: 0 Load time: 0:00.005 | Profile: Default



# In conclusione

- L'utente finale interagisce con il software di applicativo
- L'applicazione dialoga con una o più applicazioni remote utilizzando i protocolli applicativi necessari
- I protocolli applicativi sfruttano il servizio di trasporto di uno dei protocolli di trasporto per raggiungere l'applicazione remota
- Il protocollo di trasporto utilizza le capacità di instradamento di IP per la consegna dei dati al calcolatore remoto dove risiede l'applicazione
- IP consegna i dati sfruttando l'infrastruttura di rete a cui gli host sono connessi tramite l'interfaccia LAN

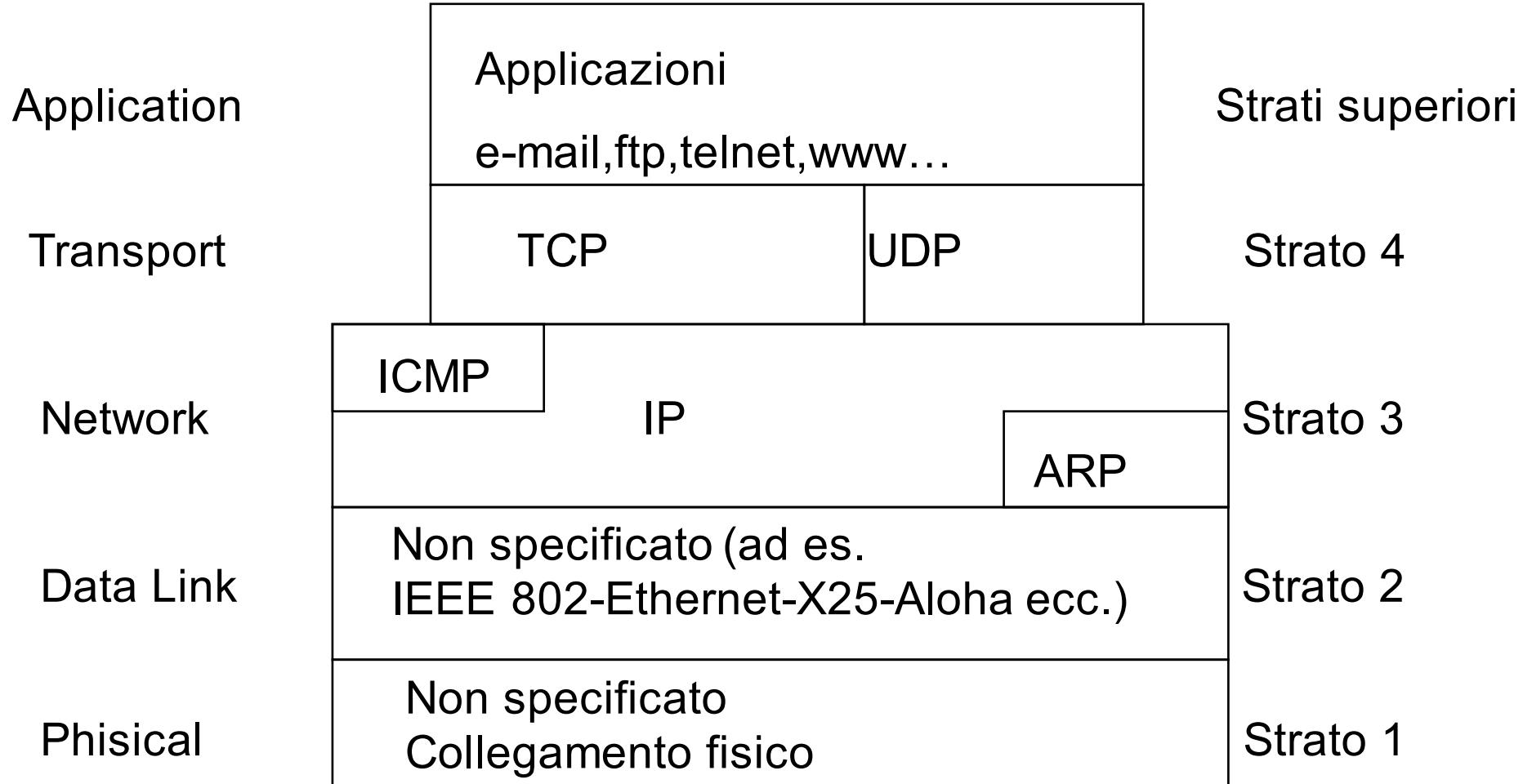


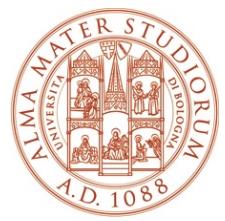
ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# I protocolli di Internet



# Architettura





# Internet Protocol (IP) - RFC 791

- Progettato per funzionare a **commutazione di pacchetto** in modalità **connectionless**
- Si prende carico della trasmissione di **datagrammi** da sorgente a destinazione, attraverso reti eterogenee
- Identifica **host** e **router** tramite indirizzi di **lunghezza fissa**, raggruppandoli in **reti IP**
- **Frammenta** e **riassembla** i datagrammi quando necessario
- Offre un servizio di tipo **best effort**, cioè non sono previsti meccanismi per
  - aumentare l' affidabilità del collegamento end-to-end,
  - eseguire il controllo di flusso e della sequenza.

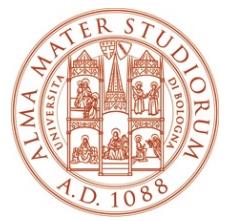


# Struttura degli indirizzi IP

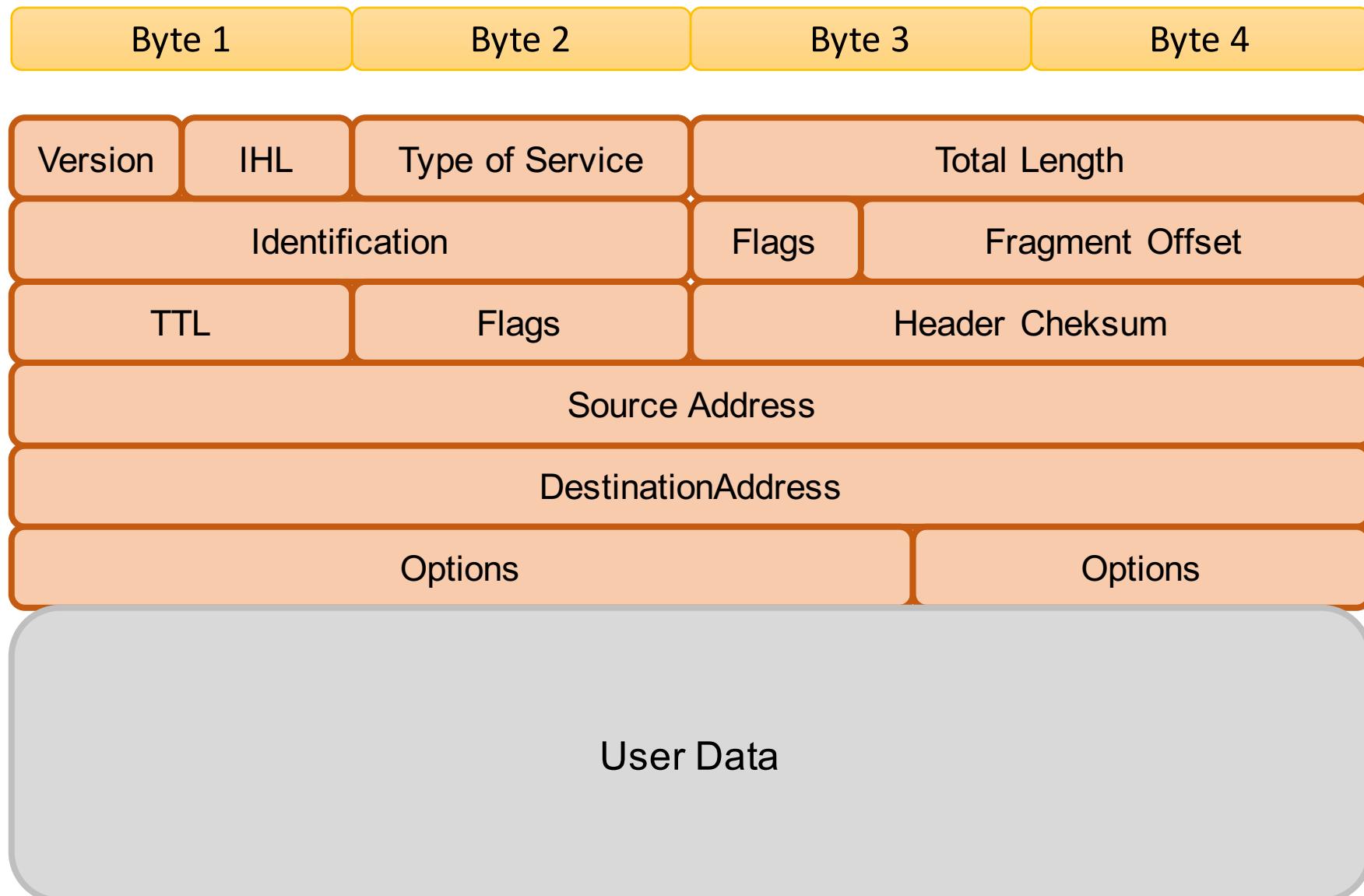
- Indirizzi di lunghezza fissa pari a **32 bit**
- Scritti convenzionalmente come sequenza di 4 numeri decimali, con valori da **0** a **255**, separati da punto (rappresentazione **dotted decimal**)

10001001.11001100.11010100.00000001  
137.204.212.1

- Numero teorico max. di indirizzi  
 **$2^{32} = 4.294.967.296$** 
  - In realtà si riesce a sfruttare un numero molto inferiore
- Assegnati dalla **IANA** (Internet Assigned Numbers Authority)



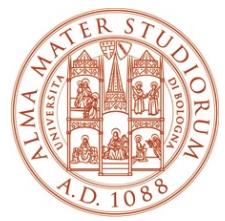
# Formato pacchetto





# Significato delle PCI

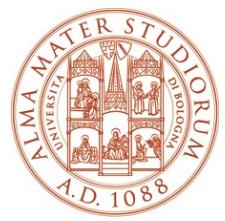
- **Version** : indica il formato dell' intestazione, attualmente la versione in uso è la 4
- **IHL** : lunghezza dell' intestazione, espressa in parole di 32 bit; lunghezza minima = 5
- **Type of service** : indicazione sul tipo di servizio richiesto, usato anche come sorta di priorità
- **Total length** : lunghezza totale del datagramma, misurata in bytes; lunghezza massima = 65535 bytes, ma non è detto che tutte le implementazioni siano in grado di gestire questa dimensione



# Significato delle PCI

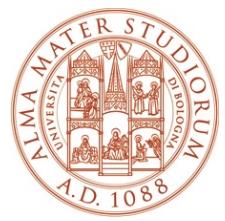
- **Identification** : valore intero che identifica univocamente il datagramma
  - Indica a quale datagramma appartenga un frammento (fragment)
- **Flag** :

|       |   |
|-------|---|
| bit 0 | sempre a 0  |
| bit 1 | don't fragment (DF)<br>DF = 0 si può frammentare<br>DF = 1 non si può frammentare |
| bit 2 | more fragments (MF)<br>MF = 0 ultimo frammento<br>MF = 1 frammento intermedio     |
- **Fragment offset**: indica quale è la posizione di questo frammento nel datagramma, come distanza in unità di 64 bit dall' inizio



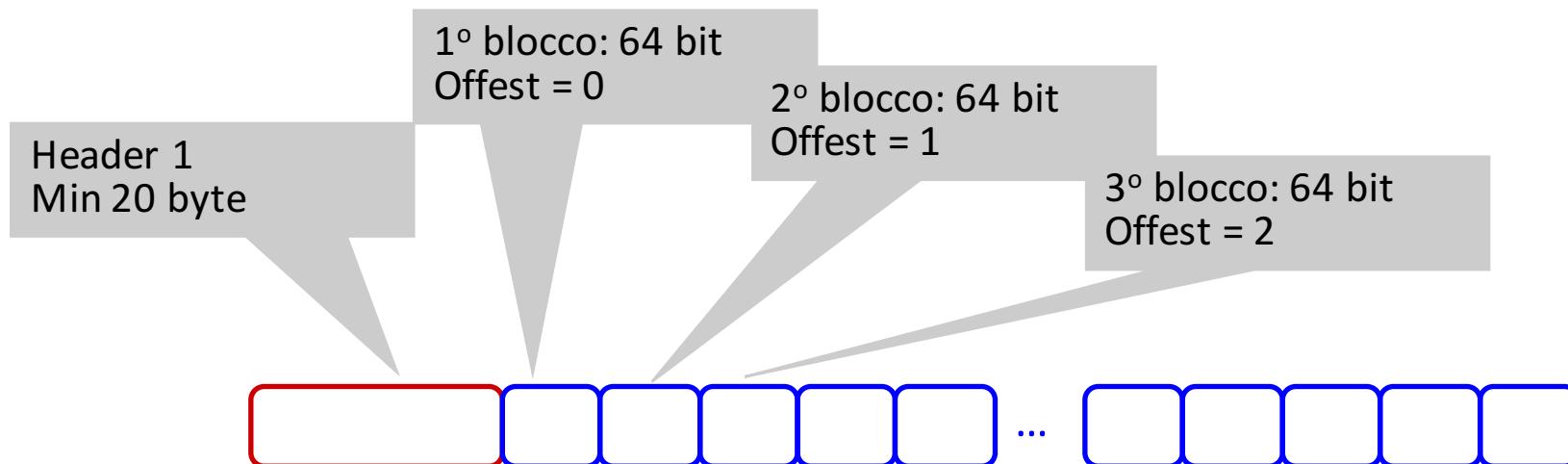
# Fragment offset

- Il datagramma IP viene virtualmente suddiviso in sotto-blocchi di 8 byte (64 bit)
- Per l'IP che trasmette (non necessariamente la sorgente dei dati ma anche un nodo intermedio)
  - Il primo blocco del datagramma è il numero 0
  - I blocchi successivi sono logicamente numerati sequenzialmente
- Il numero logico del primo blocco viene scritto nel Fragment Offset del datagramma



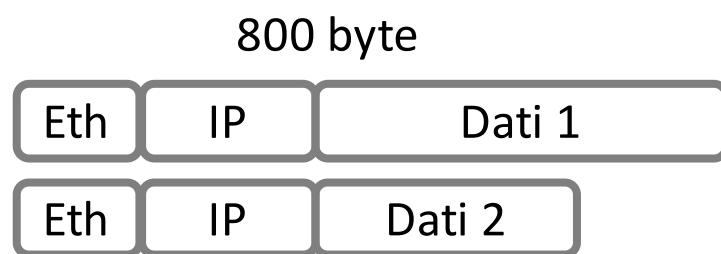
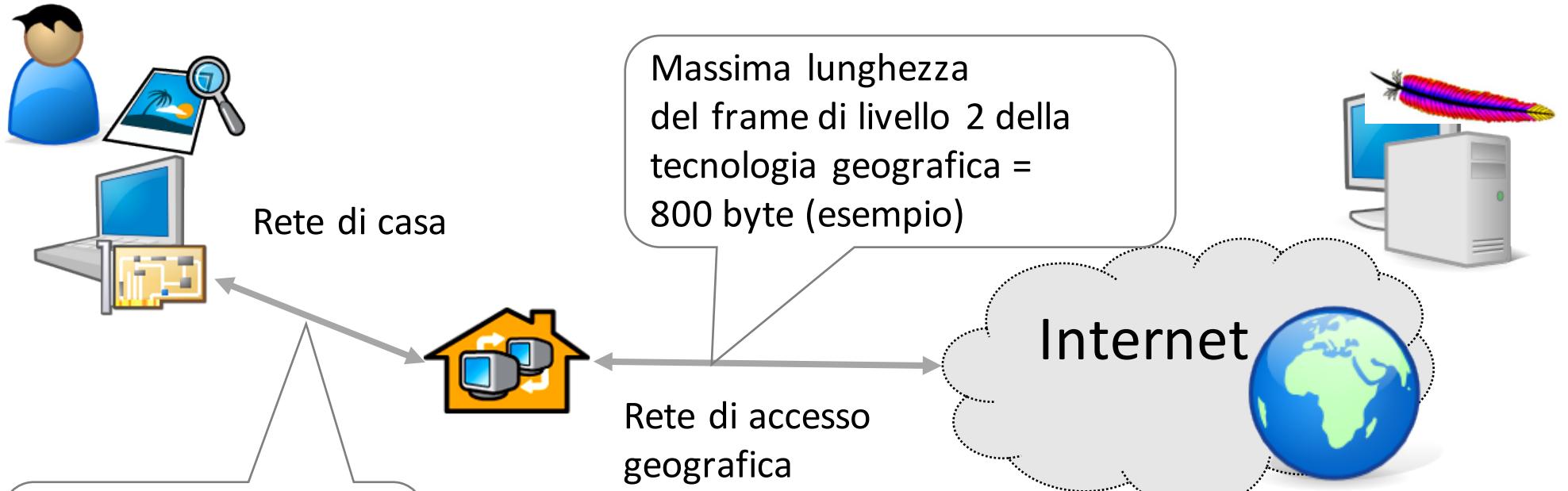
# Implementazione

- Chi frammenta i datagrammi?
  - Qualunque apparato di rete dotato di protocollo IP può frammentare un datagramma
  - Tipicamente i nodi intermedi non riassemmblano, ma lo fa solamente il terminale ricevente
- Frammentazioni multiple
  - Un datagramma può essere frammentato a più riprese in nodi successivi
- La numerazione tramite “**offset**” permette di rinumerare facilmente frammenti di un frammento



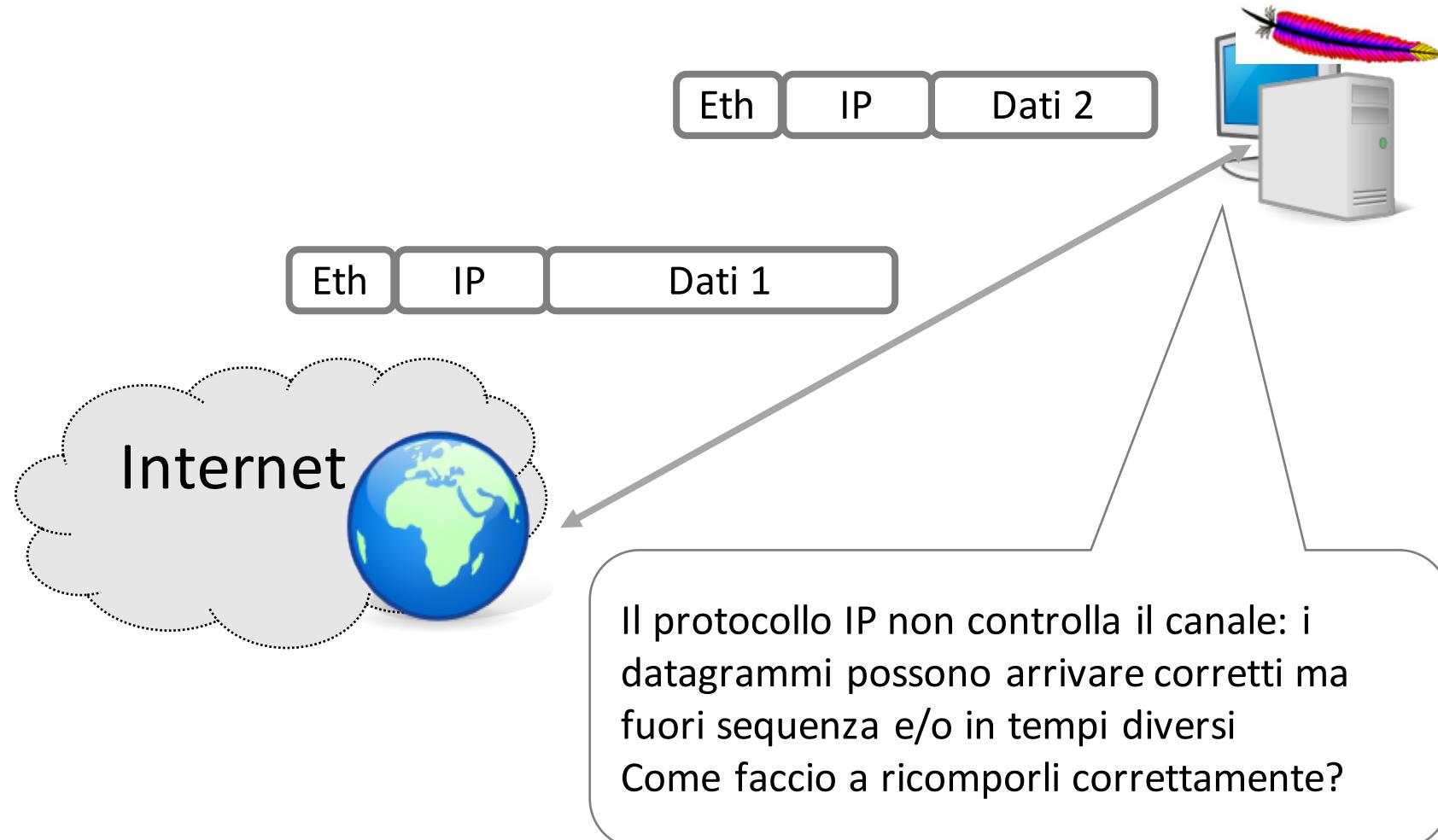


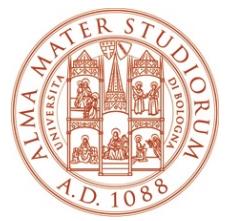
# Perché la segmentazione?



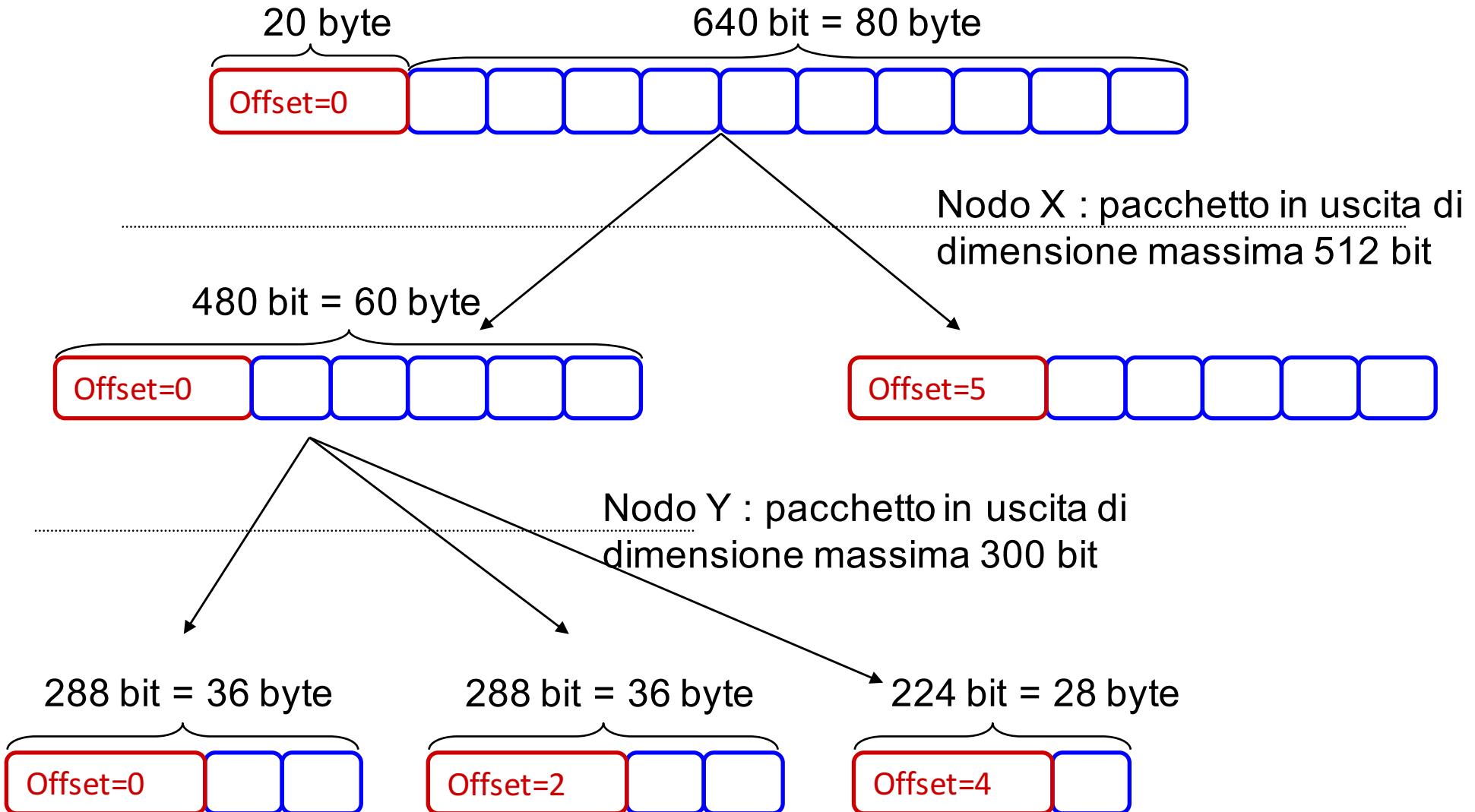


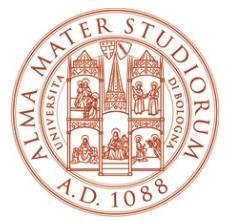
# Il riassemblamento





# Calcolo dell'offset





# Formato del pacchetto IP (4)

- **Time to live (TTL)** : max numero di nodi traversabili
  - Il nodo sorgente attribuisce un valore maggiore di 0 a TTL (tipicamente TTL = 64, al massimo 255)
  - Ogni nodo che attraversa il datagramma pone TTL = TTL - 1
  - Il primo nodo che vede TTL = 0 distrugge il datagramma
- **Protocol** : indica a quale protocollo di livello superiore appartengono i dati del datagramma
- **Header checksum** : controllo di errore della sola intestazione, viene ricalcolato da ogni nodo attraversato dal datagramma
- **Source and Destination Address** : indirizzi sorgente e destinazione



# Formato del pacchetto IP (5)

- **Options** : contiene opzioni relative al trasferimento del datagramma (registrazione del percorso, meccanismi di sicurezza), è perciò di lunghezza variabile
- **Padding** : bit privi di significato aggiunti per fare in modo che l' intestazione sia con certezza multipla di 32 bit



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# L'instradamento IP



# Instrandamento

- La rete Internet è una rete a commutazione di pacchetto
  - Oggi un sistema molto complesso
- In generale esistono più modi per raggiungere una destinazione da una certa sorgente
- Chi decide quale percorso seguire e come lo fa?
- Si decide pacchetto per pacchetto o per flusso di dati applicativi?
- ...

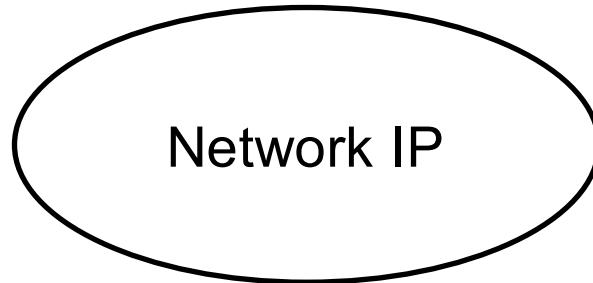


# Come funziona Internet

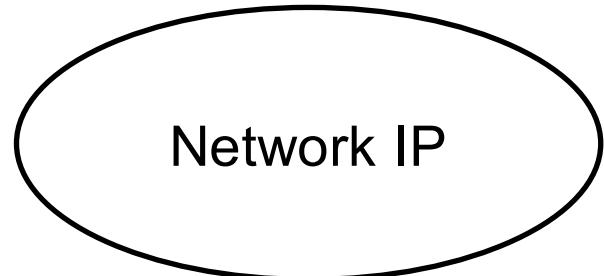
- Internet è una grande “rete di reti”
- La componente elementare è la **network IP**
  - Ogni network IP è una sorta di isola
    - L’isola tipicamente contiene calcolatori che fungono da nodi terminali della rete detti **host**
  - Le isole sono interconnesse da apparati che svolgono la funzione di “ponte”
    - Si tratta di calcolatori specializzati detti **router** o **gateway**

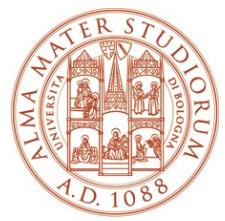


# Internet: reti di reti



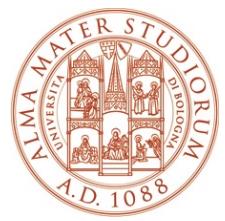
Tante Network IP isolate





# La tecnologia

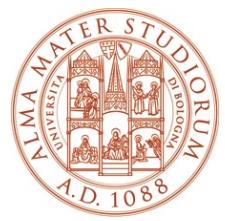
- Ogni network IP può essere implementata con una **tecnologia specifica**
- Esempio
  - Wi-Fi : Network realizzata con tecnologia wireless in area locale
  - ADSL e xDSL: Network realizzata con tecnologia a media distanza via cavo tramite infrastruttura di uno specifico fornitore di servizio pubblico
  - Ethernet: Network realizzata con tecnologia a breve distanza via cavo privata in area locale
  - GPRS/EDGE/LTE: Network realizzata con tecnologia radio a media distanza tramite infrastruttura di uno specifico fornitore di servizio pubblico



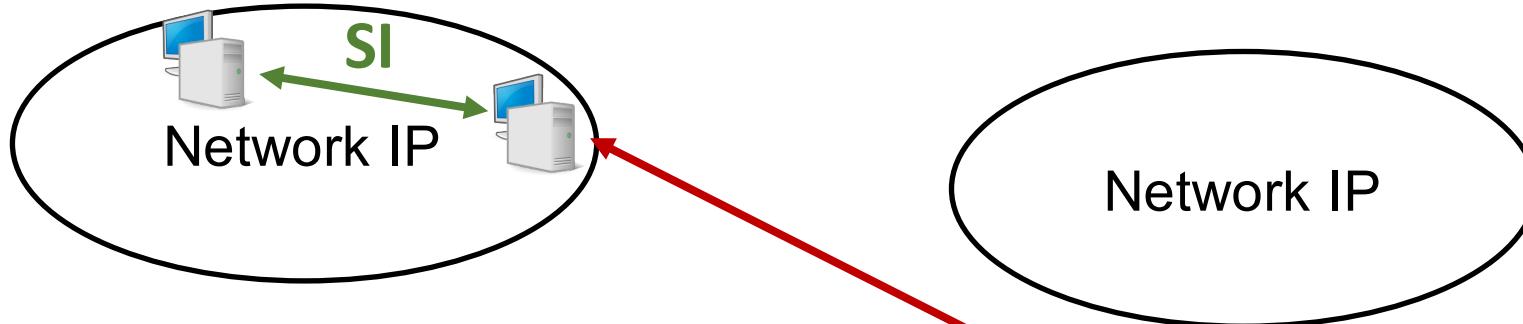
# La network IP

- I calcolatori di una network IP sono connessi dalla medesima infrastruttura di rete fisica (livelli 1 e 2)

- Ipotesi fondamentale
  - Tutti gli host appartenenti alla medesima network IP sono in grado di parlare tra loro grazie alla tecnologia con cui essa viene implementata



# Internet: reti di reti



Tante Network IP (isole) isolate

**NO**





# Rete logica e rete fisica

- Nella terminologia di Internet si definisce
  - **Rete logica**: la network IP a cui un Host appartiene logicamente
  - **Rete fisica**: la rete (tipicamente LAN) a cui un Host è effettivamente connesso
- La rete fisica normalmente ha capacità di instradamento e può avere indirizzi locali (es. indirizzi MAC)
- L'architettura a strati nasconde gli indirizzi fisici e consente alle applicazioni di lavorare solo con indirizzi IP



# Interconnettere le isole

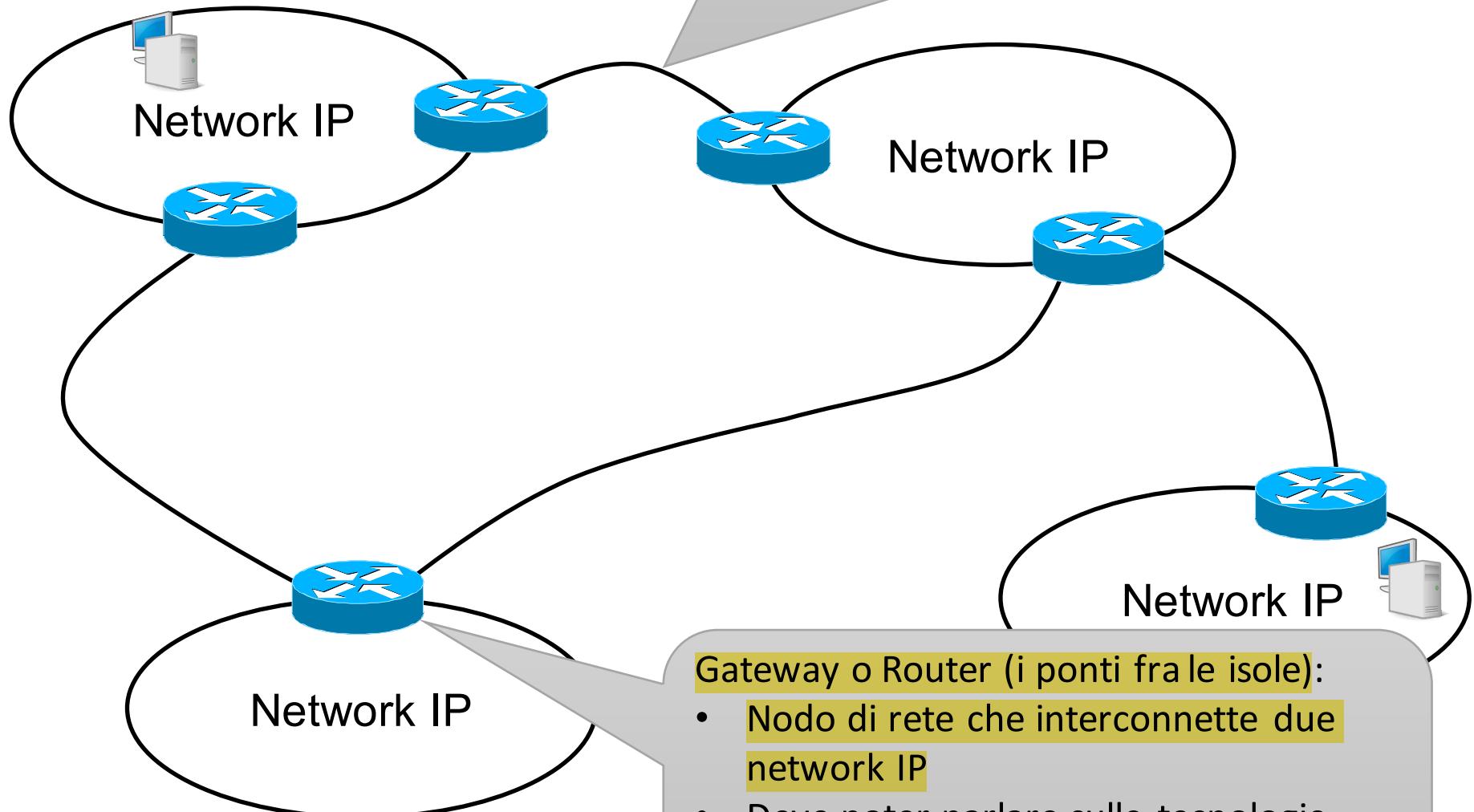
- Per far parlare tra loro le isole (network IP) è necessario che
  - Vi siano dei **collegamenti fra le isole stesse**, spesso realizzati con tecnologie diverse da quelle dell'isola
  - Vi siano degli **apparati che permettono di usare questi collegamenti nel modo opportuno**
  - Sia possibile scegliere il giusto collegamento verso l'isola che si vuole raggiungere



# I router

Collegamento fra router:

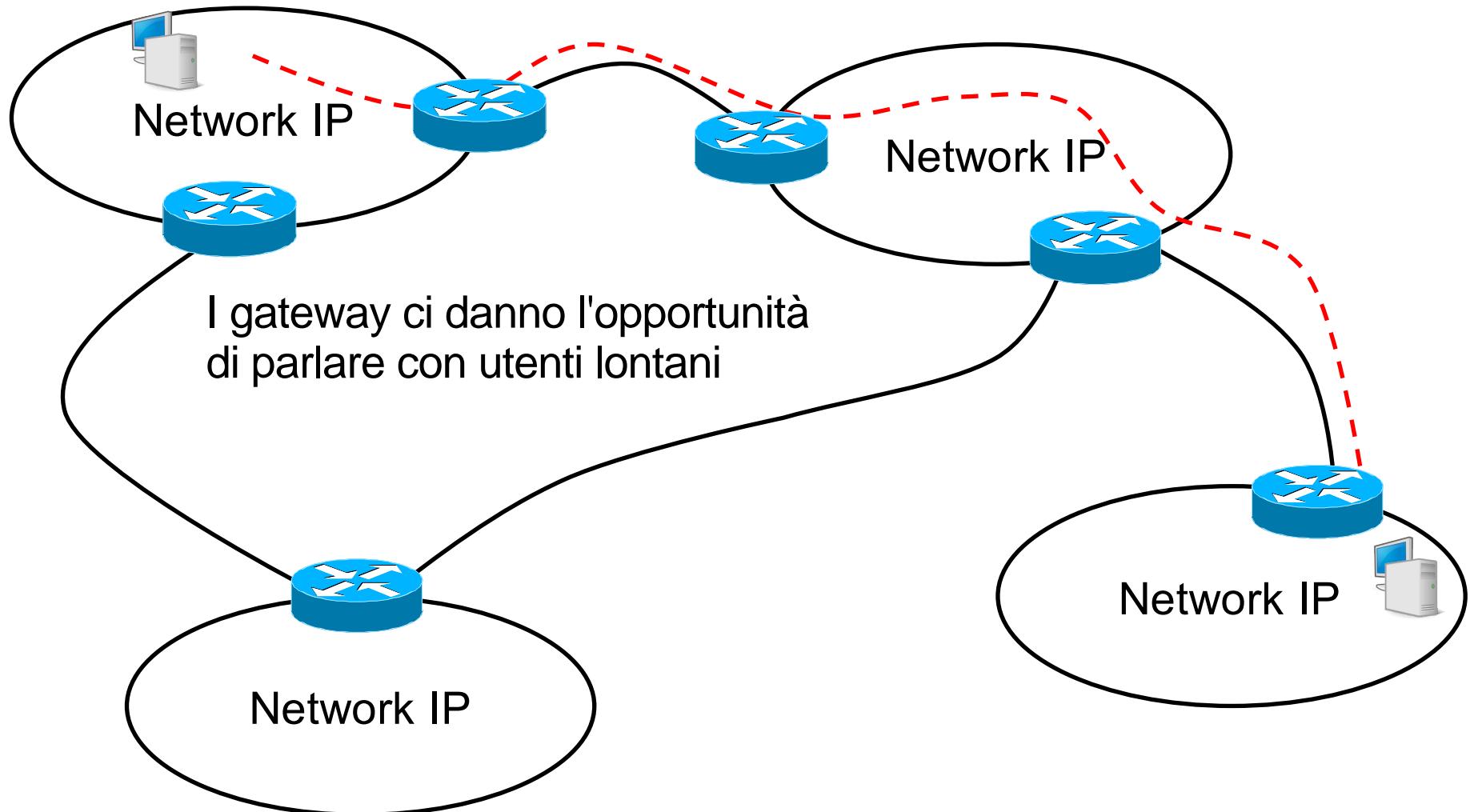
- Può essere una tecnologia simile a quella delle network oppure molto diversa

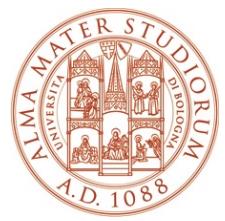


Gateway o Router (i ponti fra le isole):

- Nodo di rete che interconnette due network IP
- Deve poter parlare sulle tecnologie specifiche delle due Network
- Ha funzioni dal livello 1 al livello 3 OSI

# Il percorso end-to-end





# Cosa fa IP

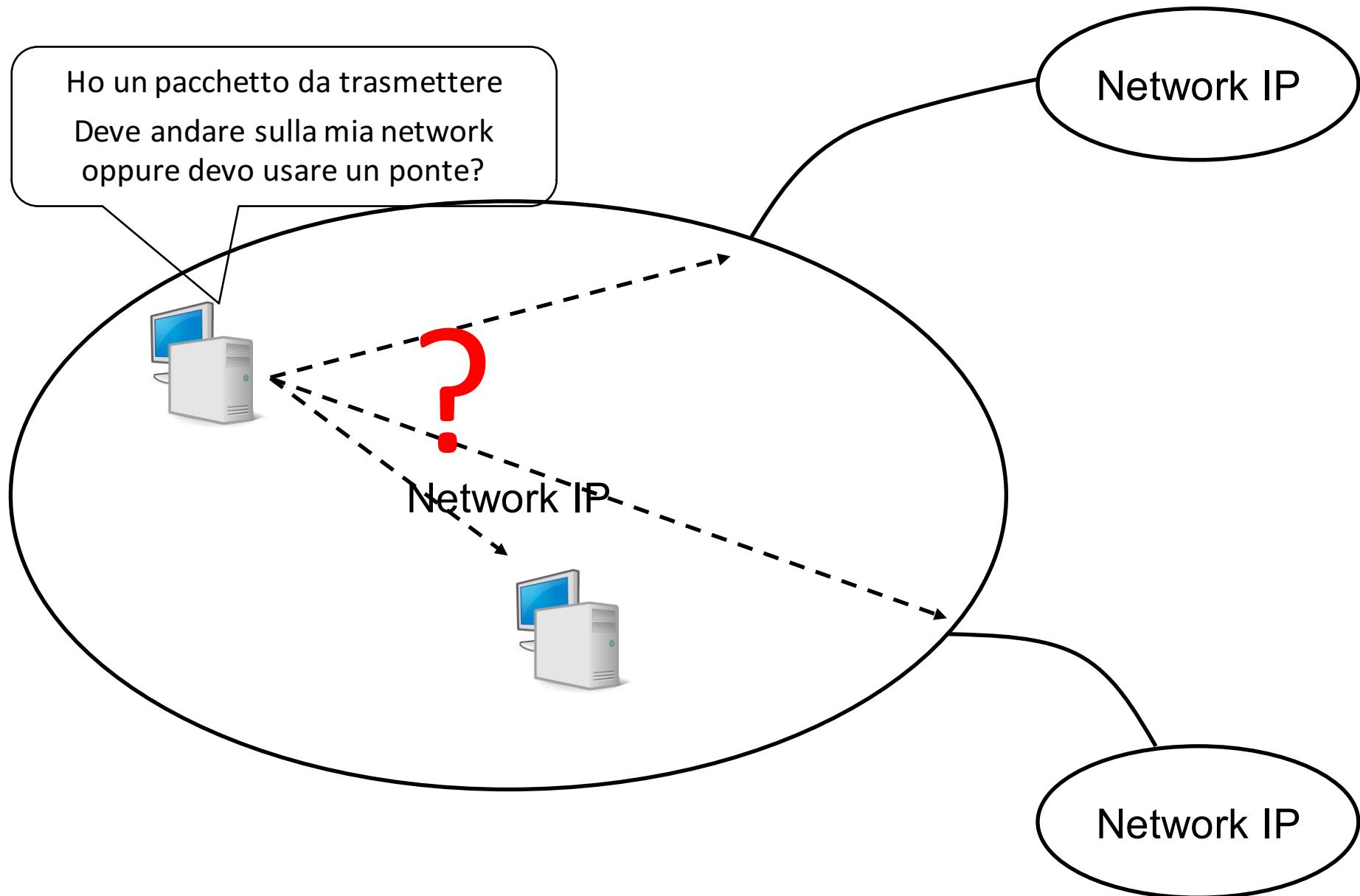
- La tecnologia IP è agnosta rispetto alla tecnologia con cui sono realizzate le network
  - Il protocollo IP è concepito per lavorare indifferentemente su tecnologie diverse

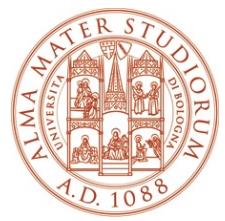
- L'obiettivo di IP è quello di rendere possibile il dialogo fra network a prescindere dalla loro implementazione e localizzazione

Protocollo di interconnessione di network



# La domanda cruciale

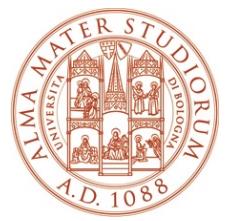




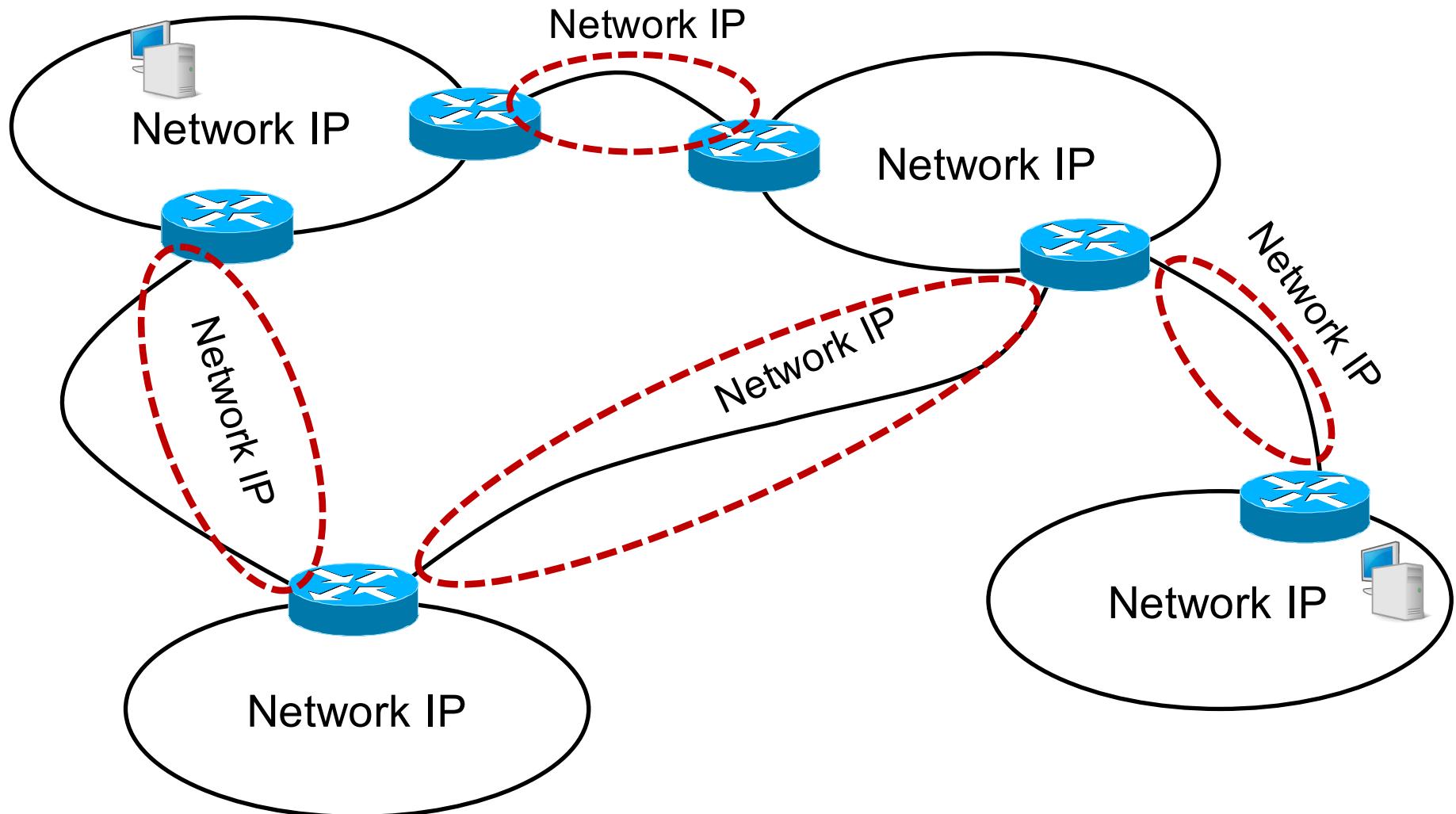
# La risposta

- Ogni nodo di Internet ha una base dati di destinazioni possibili
- Quando deve inviare un datagramma
  - Parte dall'indirizzo IP di destinazione
  - Legge la base dati
  - Decide quale azione intraprendere

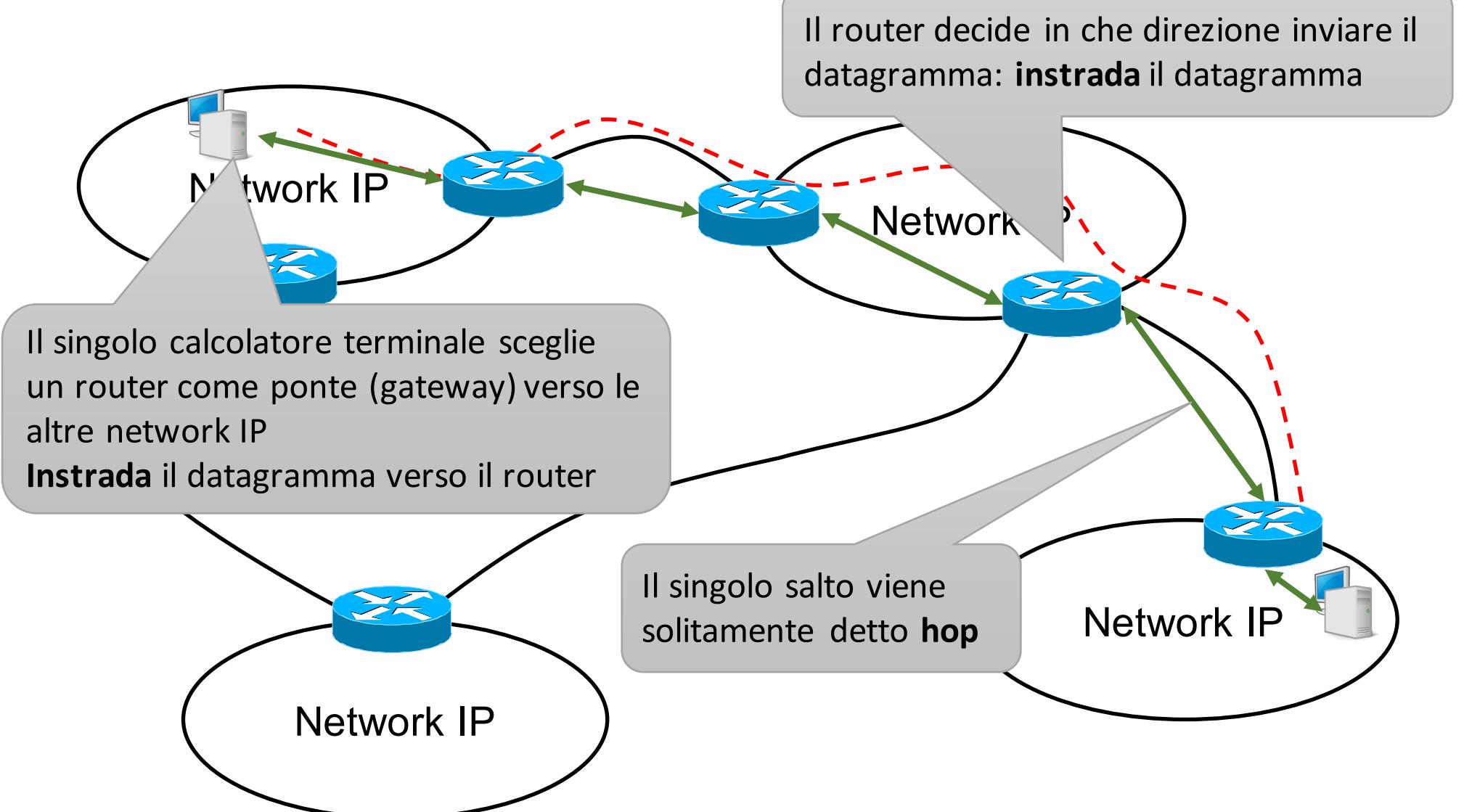
- La tecnologia della propria network può essere utilizzata:
  - Per raggiungere la destinazione finale
  - Per raggiungere il primo ponte da attraversare



# Le network fra i router



# L'instradamento IP





# Semantica dell'indirizzo IP

- L' indirizzo IP è logicamente suddiviso in due parti:
  - **Network (Net) ID**
    - Prefisso che identifica la **Network IP** a cui appartiene l' indirizzo
    - Tutti gli indirizzi di una medesima **Network IP** hanno il medesimo **Network ID**
  - **Host ID**
    - Identifica l' host (l' interfaccia) vero e proprio di una certa Network
- Per Net e Host ID vengono utilizzati bit contigui
  - Net ID occupa la parte *sinistra* dell' indirizzo
  - Host ID occupa la parte *destra* dell' indirizzo



# Reti IP private (RFC 1918)

- Alcuni gruppi di indirizzi sono riservati a reti IP private
  - Essi non sono raggiungibili dalla rete pubblica
  - I router di Internet non instradano datagrammi destinati a tali indirizzi
  - Possono essere riutilizzati in reti isolate
- 
- da **10.0.0.0** a **10.255.255.255**
  - da **172.16.0.0** a **172.31.255.255**
  - da **192.168.0.0** a **192.168.255.255**



# Come si distingue net-ID da host-ID?

- Si usa la netmask
  - Al numero IP viene associata una **maschera** di 32 bit

137.204.191.25  
10001001.11001100.10111111.00011001  
11111111.11111111.11111111.11000000

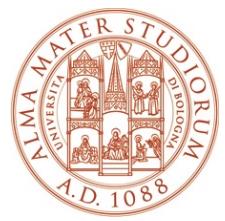
|        |         |
|--------|---------|
| Net-ID | Host-ID |
|--------|---------|

- I bit a 1 della netmask identificano i bit dell' indirizzo IP che fanno parte del net-ID
- La netmask si può rappresentare
  - In notazione dotted-decimal
    - 11111111.11111111.11111111.11000000 = 255.255.255.192
  - In notazione esadecimale
    - 11111111.11111111.11111111.11000000 = ff.ff.ff.c0
  - Utilizzando la notazione abbreviata
    - 11111111.11111111.11111111.11000000 = /26



# Netmask

- Esempio:
  - Network 192.168.1.0
    - Network privata con Net-ID = 3 byte = 24 bit
  - Subnetting in 2 sottoreti
    - Net-ID+subnet-ID = 25 bit
    - Netmask = 11111111. 11111111.  
11111111.10000000
  - Notazione
    - Net-ID = 192.168.1.0 Netmask = 255.255.255.128
    - Net-ID = 192.168.1.128 Netmask = 255.255.255.128
      - oppure
    - 192.168.1.0/25
    - 192.168.1.128/25



# Esempio: Università di Bologna

- **Net ID = 137.204**
  - La network corrispondente ha indirizzo **137.204.0.0**
  - Tutti i numeri IP dell' Università di Bologna hanno il medesimo prefisso
- **Host ID**
  - Qualunque combinazione dei rimanenti 16 bit
    - Escluso 137.204.0.0 e 137.204.255.255
  - Server web UniBO
    - 137.204.24.35
  - Server web del DEIS
    - 137.204.24.40
  - Server web DEISNet
    - 137.204.57.85



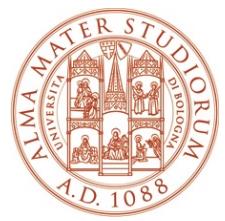
# La tabella di instradamento IP

- Base dati in forma di tabella
  - Righe (dette anche route, rotte, entry, record)
    - Insieme di informazioni relative alla singola informazione di instradamento
  - Colonne (dette campi)
    - Informazioni del medesimo tipo relative a diverse opzioni di instradamento
- Formato della tabella
  - Dipende dal sistema operativo e dall'implementazione
    - Le informazioni sono le medesime
    - Il modo di presentarle ed elaborarle può essere diverso



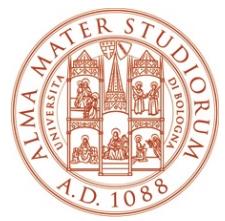
# Route

- Tipici campi della singola rotta sono:
  - **Destinazione (D)**: numero IP valido
    - Può essere un indirizzo di network o di host
  - **Netmask (N)**: maschera di rete valida
    - Identifica il Net-ID
  - **Gateway (G)**: numero IP a cui consegnare il datagramma
    - Indica il tipo di consegna da effettuare
  - **Interfaccia di rete (IF)**: interfaccia di rete utilizzare (loopback compreso) per la consegna del datagramma
    - Seleziona il dispositivo hardware da utilizzare per l' invio del datagramma
  - **Metrica (M)**: specifica il “costo” di quel particolare route
    - Possono esistere più route verso una medesima destinazione



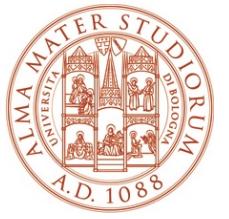
# La tabella

| Destination  | Netmask         | Gateway        | Interface | Metric |
|--------------|-----------------|----------------|-----------|--------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      | 1      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       | 1      |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       | 1      |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en2       | 1      |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       | 1      |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      | 1      |



# Uso della tabella di routing

- Il singolo nodo riceve un datagramma:
  - Estrae dall' intestazione IP\_D = indirizzo IP di destinazione
  - Seleziona il route per tale IP\_D, confrontandolo con i campi D presenti nella tabella
    - Processo di “**table lookup**”
  - Se il route esiste
    - Esegue l'azione di instradamento suggerita dai campi G e IF
  - Se il route non esiste genera un messaggio di errore
    - Tipicamente notificato all' indirizzo sorgente (ICMP - **Destination Unreachable**)



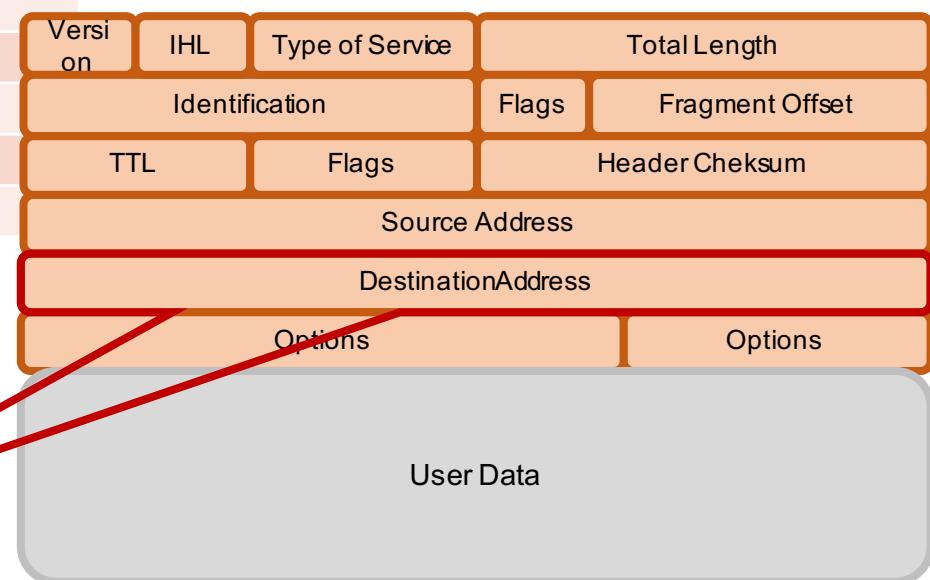
# Table lookup

- La ricerca nella tabella avviene confrontando
  - Indirizzo IP di destinazione **IP\_D** del datagramma
  - Destinazione (D) di ciascun route
  - Utilizzando la **netmask (N)** del route
- La procedura viene detta di “longest prefix match”
  - **IP\_D AND N = R**
    - Indirizzo di destinazione del datagramma e netmask di ciascuna riga
  - **R = D ?**
    - SI : la route viene selezionata e il processo termina
    - NO : si passa al route successivo
- In quale ordine leggere i route
  - dalla riga che presenta una netmask con un numero maggiore di bit a uno



# II lookup

| Destination  | Netmask         | Gateway        | Interface | Metric |
|--------------|-----------------|----------------|-----------|--------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      | 1      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       | 1      |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       | 1      |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en2       | 1      |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       | 1      |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      | 1      |



Destination Address  
Netmask **AND**

**Result**  
==  
Destination

**YES/NO**



# Esempio di lookup – 1

|   | Destinazione | Netmask         | Etc. |
|---|--------------|-----------------|------|
| 1 | 0.0.0.0      | 0.0.0.0         | ...  |
| 2 | 192.168.2.0  | 255.255.255.0   | ...  |
| 3 | 192.168.2.18 | 255.255.255.255 | ...  |

- Datagramma con IP dest. = 192.168.2.18
  - Confronto prima con riga 3, poi con riga 2 e poi riga 1

192.168.002.018      bitwise AND  
255.255.255.255  
192.168.002.018 == 192.168.002.018

- La riga 3 è quella giusta (host specific)



# Esempio di lookup – 2

|   | Destinazione | Netmask         | Etc. |
|---|--------------|-----------------|------|
| 1 | 0.0.0.0      | 0.0.0.0         | ...  |
| 2 | 192.168.2.0  | 255.255.255.0   | ...  |
| 3 | 192.168.2.18 | 255.255.255.255 | ...  |

- Datagramma con IP dest. = 192.168.2.22

192.168.002.022

255.255.255.255

192.168.002.022 != 192.168.002.018

192.168.002.022

255.255.255.000

192.168.002.000 == 192.168.002.000

- La riga 2 è quella giusta (network specific)



# Esempio di lookup – 3

|   | Destinazione | Netmask         | Etc. |
|---|--------------|-----------------|------|
| 1 | 0.0.0.0      | 0.0.0.0         | ...  |
| 2 | 192.168.2.0  | 255.255.255.0   | ...  |
| 3 | 192.168.2.18 | 255.255.255.255 | ...  |

- Datagramma con IP dest. = 80.48.15.170

080.048.015.170  
255.255.255.255  
080.048.015.170 != 192.168.002.018

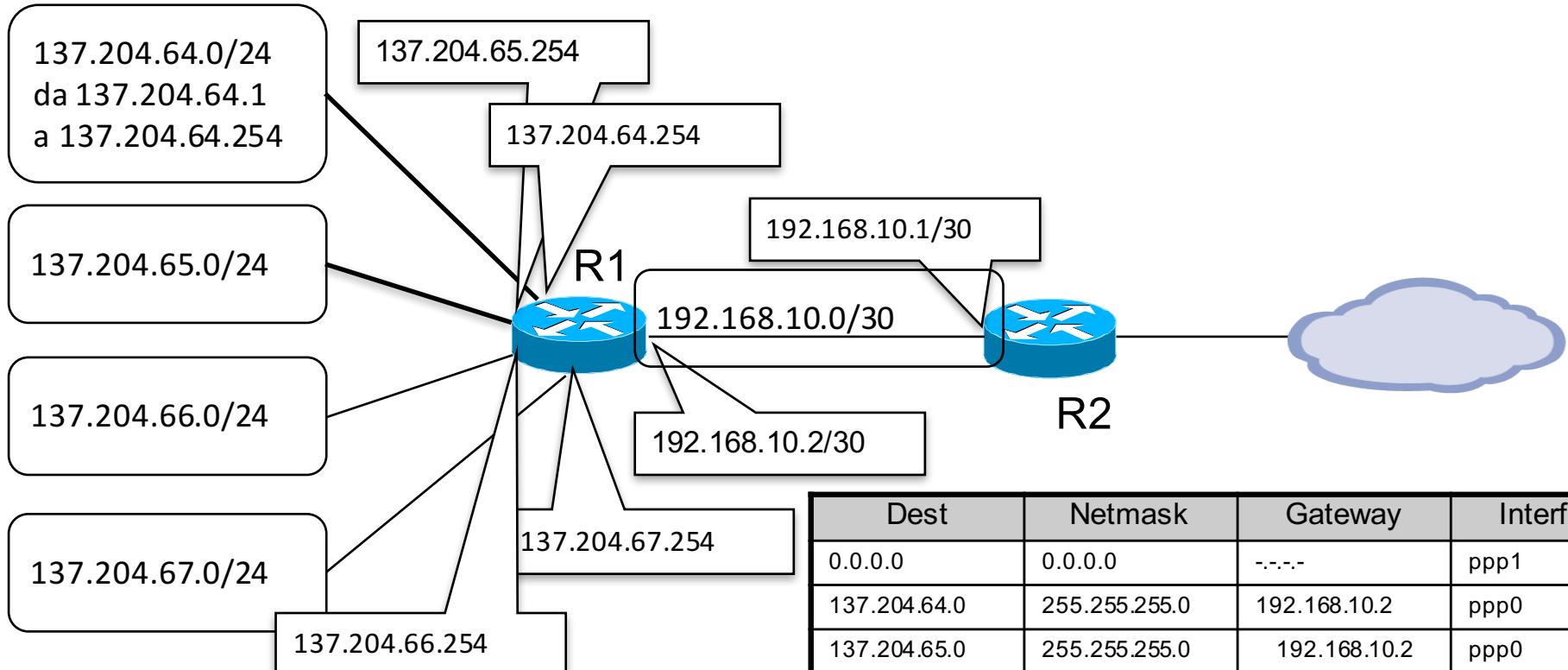
080.048.015.170  
255.255.255.000  
080.048.015.000 != 192.168.002.000

080.048.015.170  
000.000.000.000  
000.000.000.000 == 000.000.000.000

- La riga 1 è quella giusta (default gateway)



# Esempio



| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en2       |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      |



# Analizziamo gli indirizzi delle 4 reti

- 137.204.64.0 il terzo byte è 01000000
- 137.204.65.0 il terzo byte è 01000001
- 137.204.66.0 il terzo byte è 01000010
- 137.204.67.0 il terzo byte è 01000011
  - I primi 2 byte ed i primi 6 bit del terzo byte sono comuni a tutte e quattro le network. Se usiamo NETMASK=255.255.252.0

10001001.11001100.01000000.xxxxxxxx  
11111111.11111111.11111100.00000000  
10001001.11001100.01000000.00000000

10001001.11001100.01000001.xxxxxxxx  
11111111.11111111.11111100.00000000  
10001001.11001100.01000000.00000000

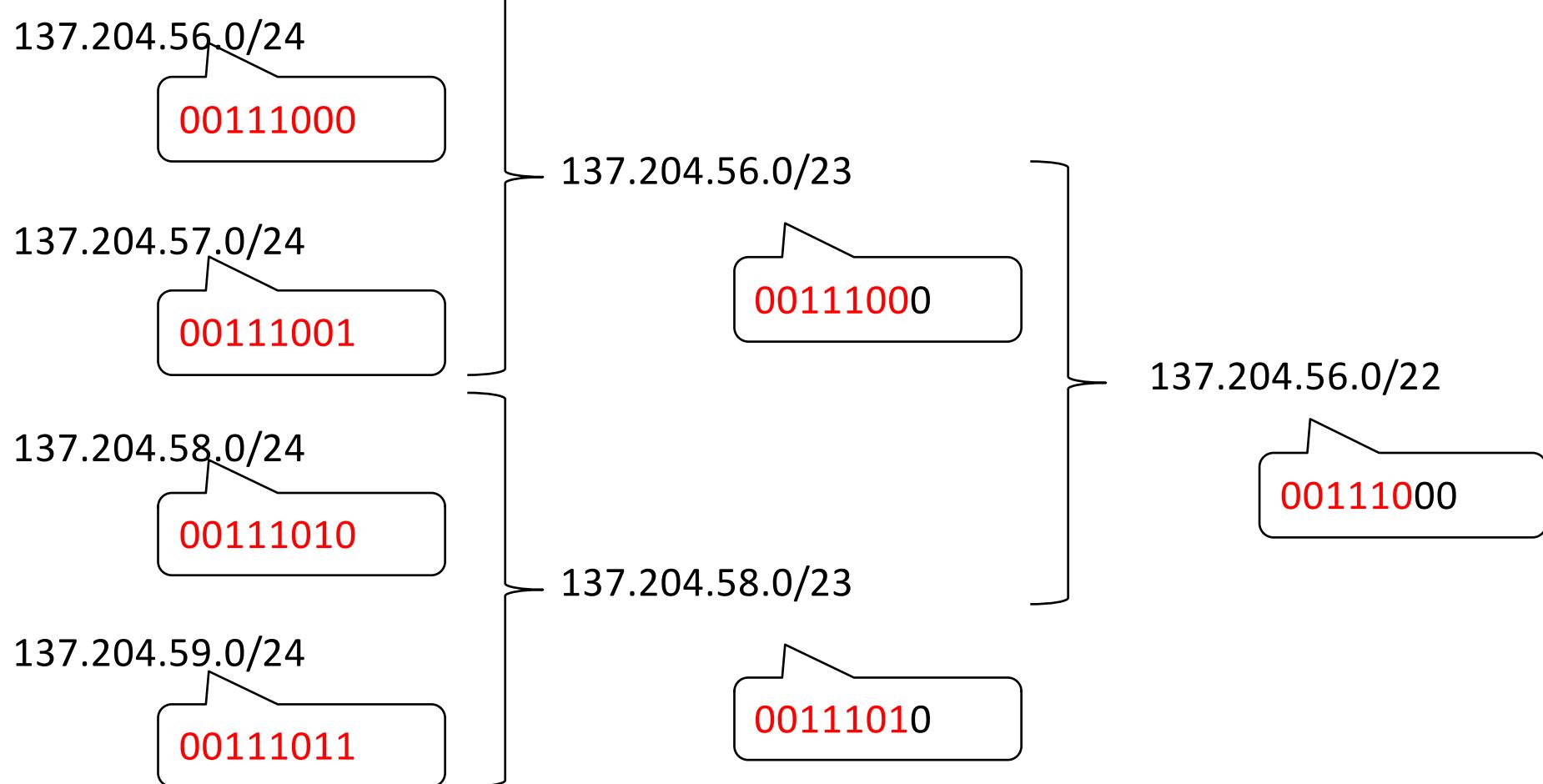
10001001.11001100.01000010.xxxxxxxx  
11111111.11111111.11111100.00000000  
10001001.11001100.01000000.00000000

10001001.11001100.01000011.xxxxxxxx  
11111111.11111111.11111100.00000000  
10001001.11001100.01000000.00000000

- Otteniamo il medesimo risultato in tutti e quattro i casi:
  - Il prefisso di rete è sempre 137.204.64.0



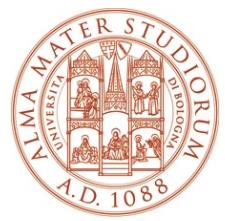
# Un altro esempio



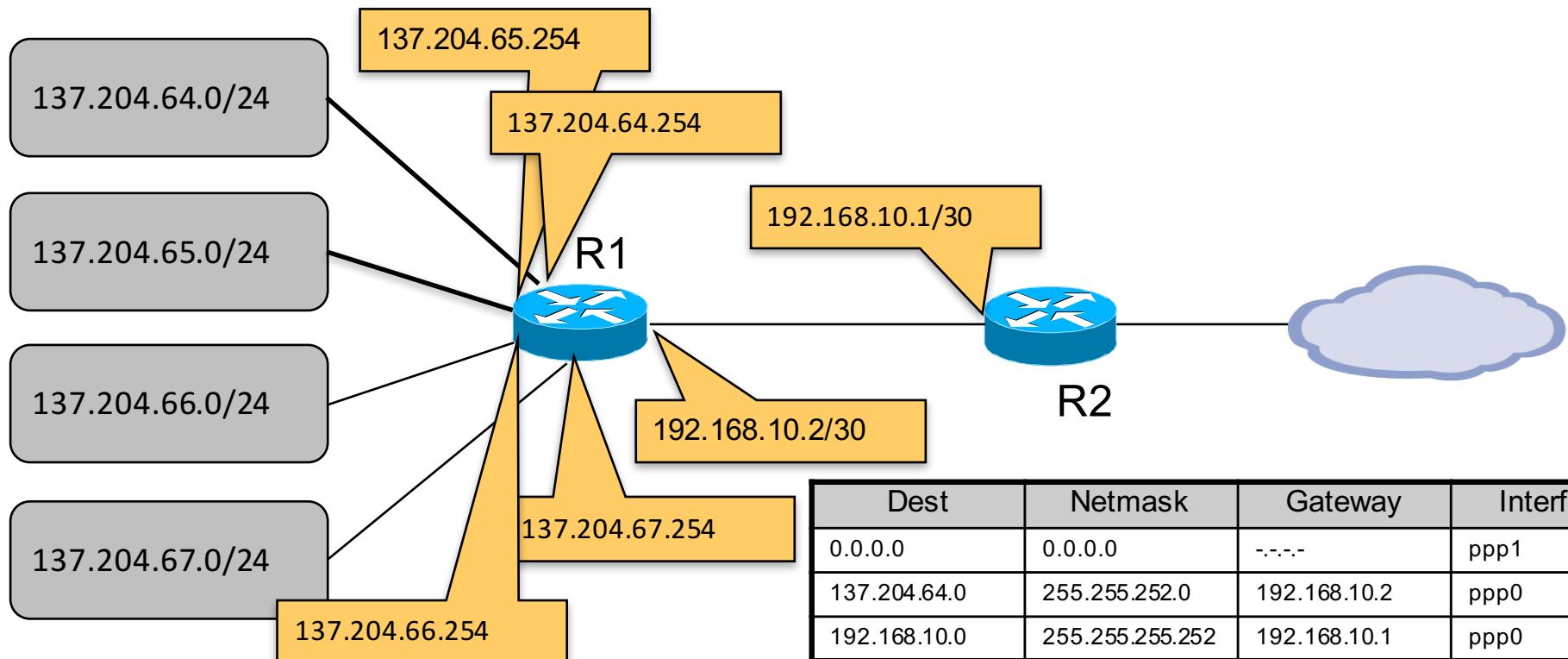


# Semplificazione delle tabelle

- È necessario che R2 conosca il dettaglio di come le reti sono connesse a R1?
  - R2 invia comunque i datagrammi tramite R1
  - È sufficiente un'informazione più “riassuntiva”
- I route verso le 4 network possono essere aggregate in una sola
- R2 vede le 4 reti come una sola
  - Il gateway verso quelle destinazioni è R1



# Aggregazione



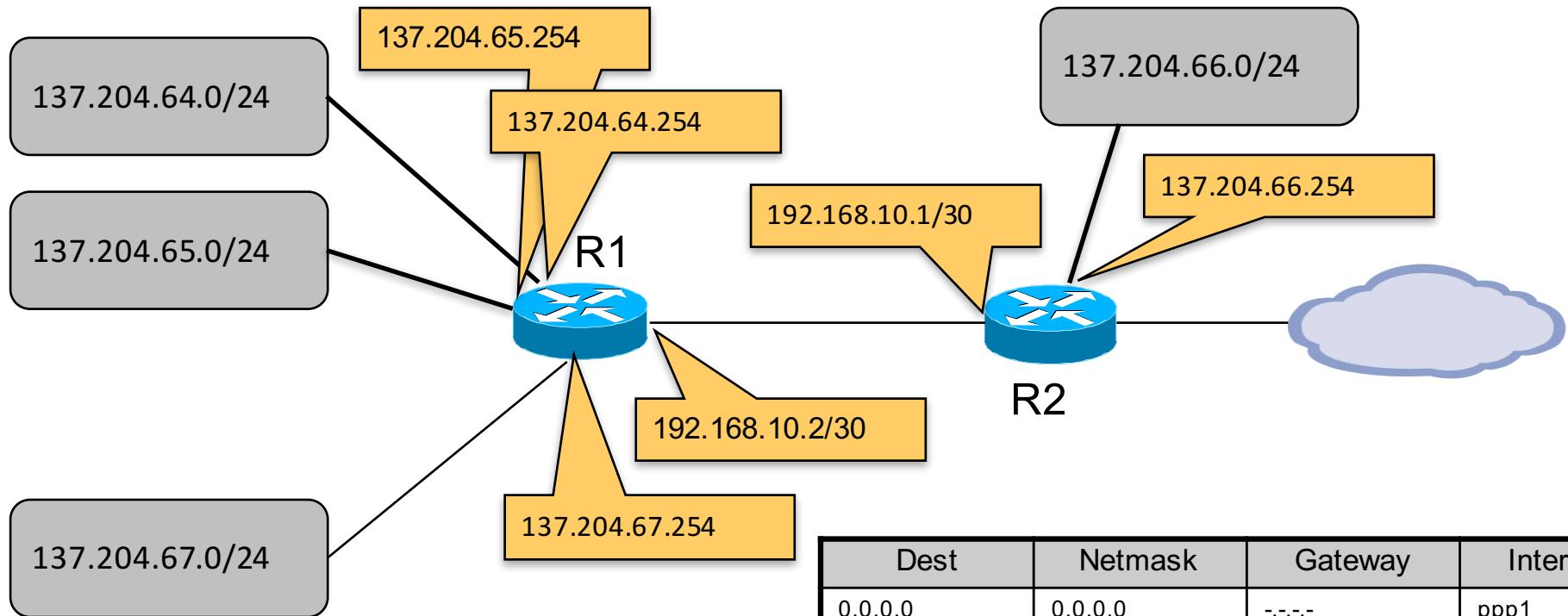
| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en2       |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      |



# Perché ordinare i route?

- Dare priorità alle route più specifiche
- L'ordinamento in funzione della Netmask decrescente garantisce di considerare in ordine
  - singoli host
  - reti piccole
  - reti grandi
- È possibile implementare eccezioni a regole generali che possono convivere nella medesima tabella

# Eccezioni

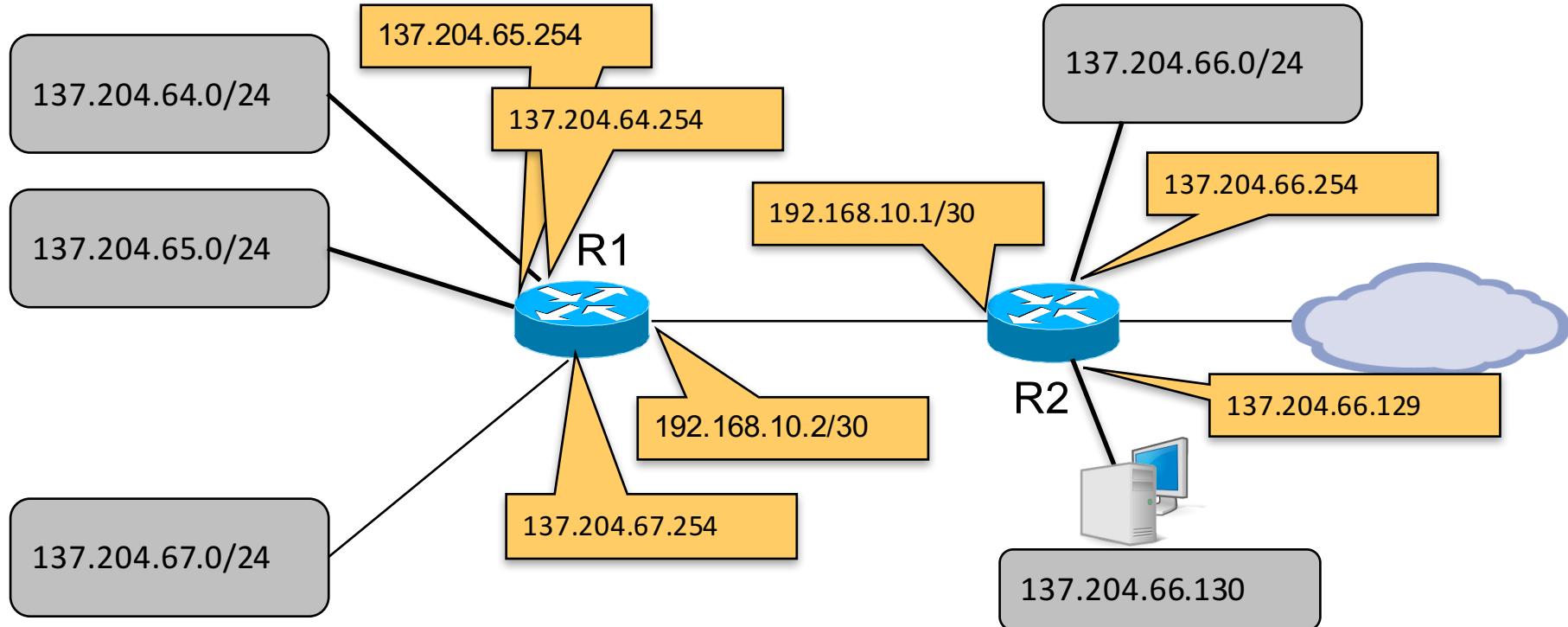


| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | -.-.-          | ppp1      |
| 137.204.64.0 | 255.255.252.0   | 192.168.10.2   | ppp0      |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en0       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.1   | Ppp0      |

| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      |



# Eccezioni



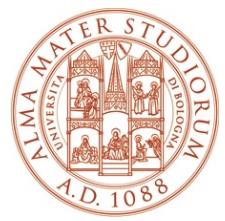
| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      |

| Dest           | Netmask         | Gateway        | Interface |
|----------------|-----------------|----------------|-----------|
| 0.0.0.0        | 0.0.0.0         | -.-.-          | ppp1      |
| 137.204.64.0   | 255.255.252.0   | 192.168.10.2   | ppp0      |
| 137.204.66.0   | 255.255.255.0   | 137.204.66.254 | en0       |
| 192.168.10.0   | 255.255.255.252 | 192.168.10.1   | Ppp0      |
| 137.204.66.130 | 255.255.255.255 | 137.204.66.129 | en1       |



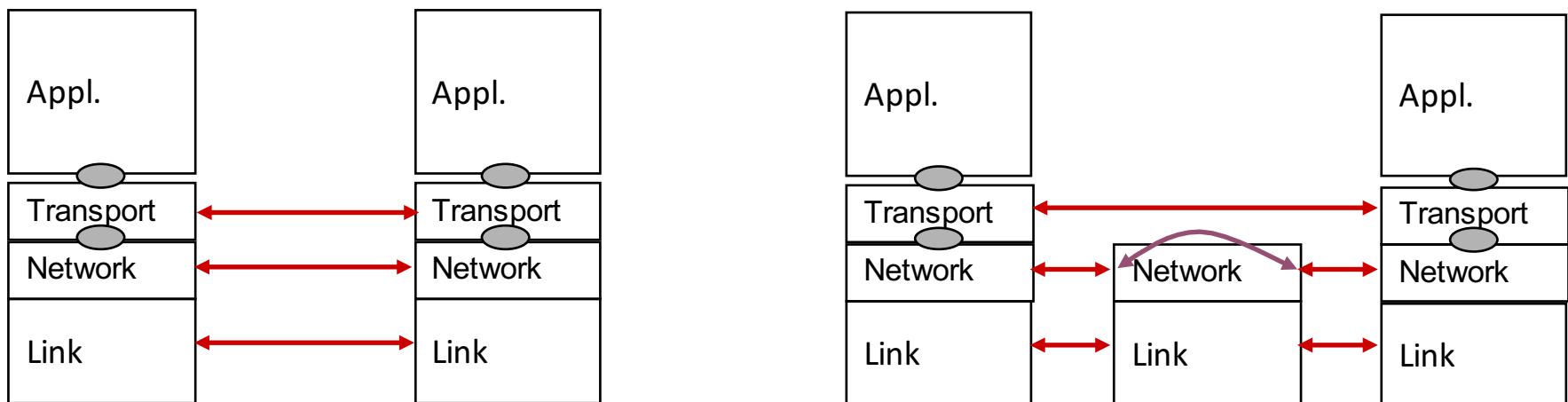
# Gateway

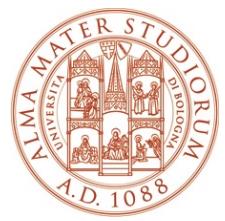
- Nella tabella di instradamento compaiono
  - Gateway
  - Interfaccia
- Perché due informazioni distinte?
- Chi è il gateway?



# Il ruolo del Gateway

- Il table look-up sceglie la D i-esima =  $D_i$
- La funzione di instradamento invia il datagramma a  $IF_i$
- Con l' obiettivo di consegnarlo al **gateway**  $G_i$
- Perché non è sufficiente  $IF_i$ ?
- Normalmente l' instradamento IP è basato sulle network
  - Host della medesima network possono comunicare direttamente
  - Host di network diverse comunicano tramite un router
- **Gateway** = responsabile della consegna del datagramma



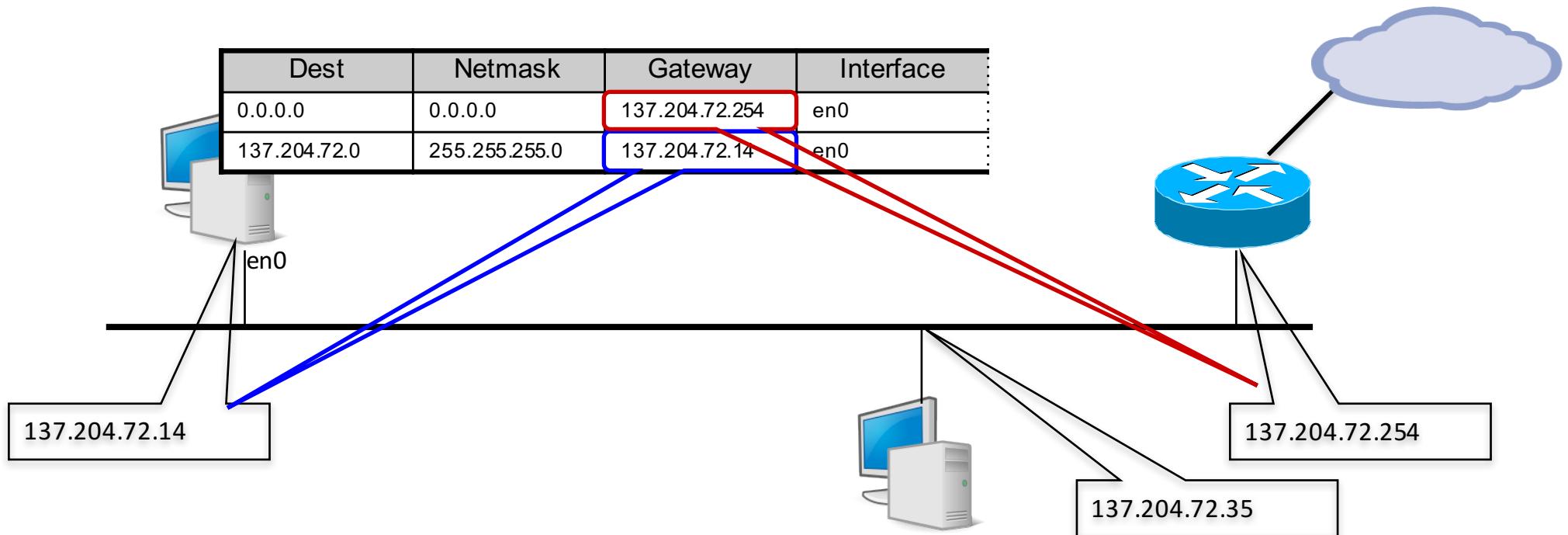


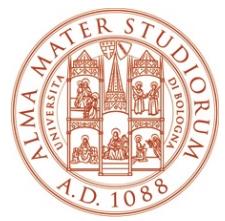
# Instradamento diretto e indiretto

- **Routing** : scelta del percorso su cui inviare i dati
  - i router formano struttura interconnessa e cooperante:
    - i datagrammi passano dall' uno all' altro finché raggiungono quello che può consegnarli direttamente al destinatario
- **Direct delivery** :
  - IP sorgente e IP destinatario sono sulla stessa rete fisica
  - L' host sorgente spedisce il datagramma direttamente al destinatario
- **Indirect delivery** :
  - IP sorgente e IP destinatario non sono sulla stessa rete fisica
  - L' host sorgente invia il datagramma ad un router intermedio

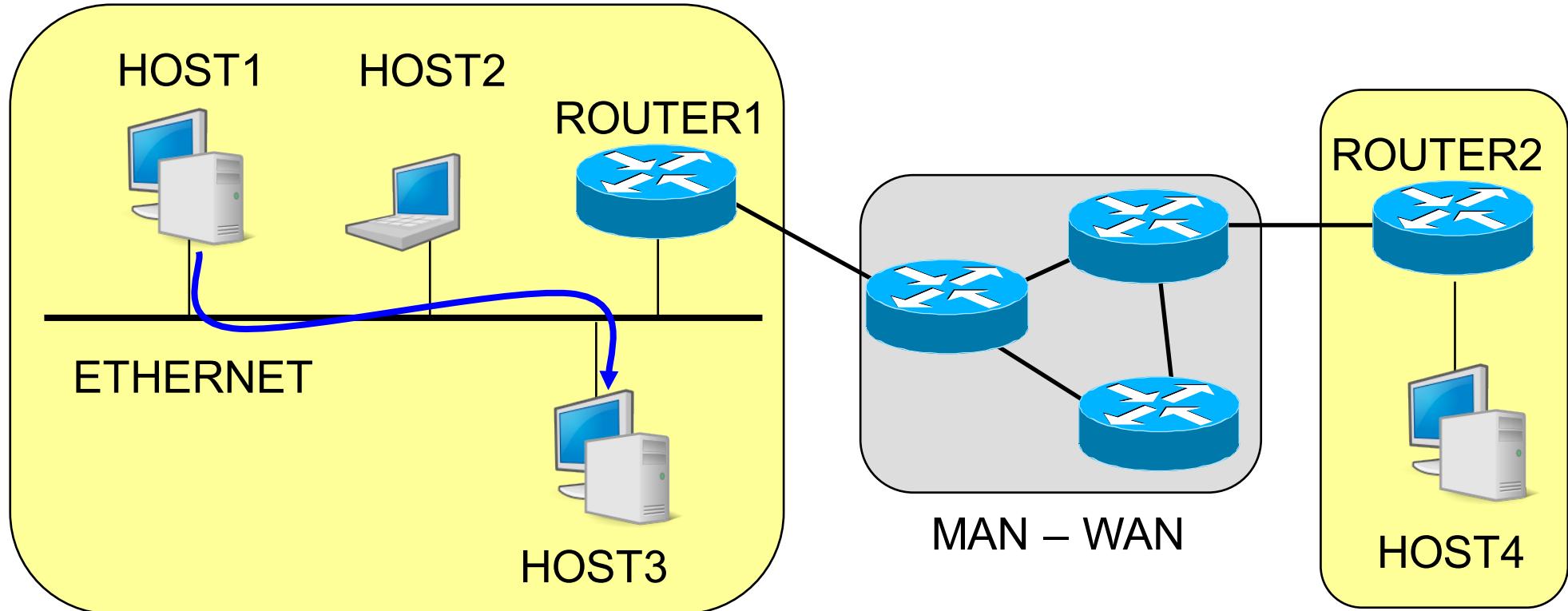
# Uso del Gateway

- Il campo gateway della tabella di routing serve per specificare il tipo di instradamento
  - Instradamento diretto: la sintassi dipende dall' implementazione
    - In Windows: instradamento diretto se gateway = IP locale
    - In Linux/Unix: instradamento diretto se gateway = 0.0.0.0
  - Instradamento indiretto
    - Gateway = numero IP del router da contattare





# Direct Delivery



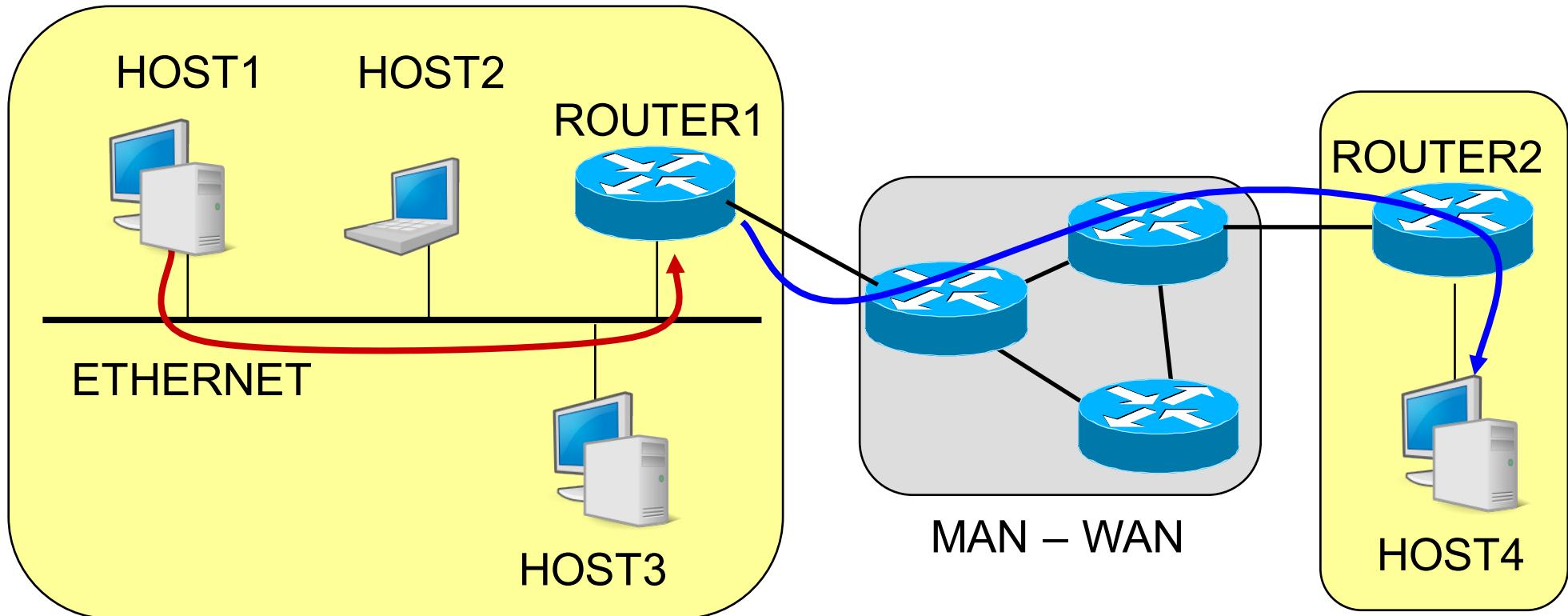
L2 ADDRESS: HOST3

IP ADDRESS: HOST3

DATI



# Indirect Delivery



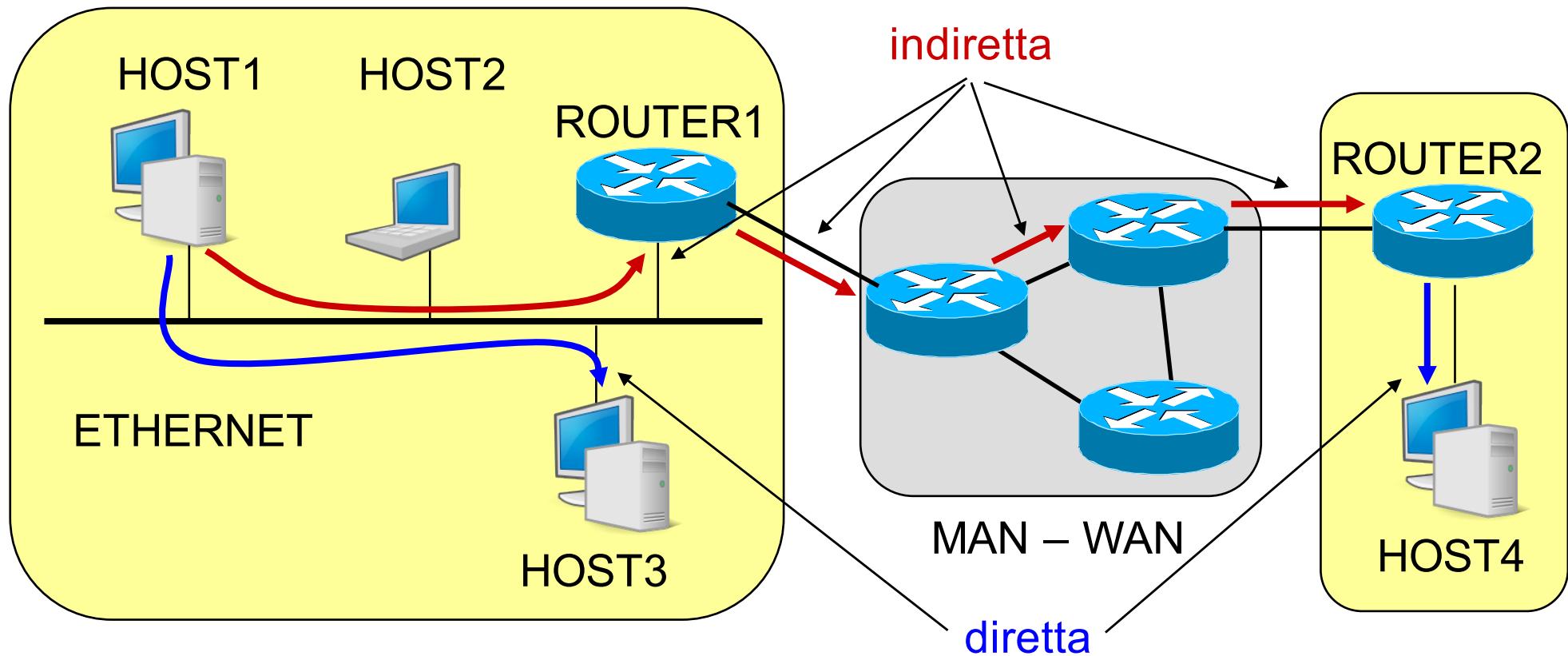
L2 ADDRESS: ROUTER1

IP ADDRESS: HOST4

DATI

# Da mittente a destinatario

- C' è sempre una consegna diretta
- Può non esserci alcuna consegna indiretta
- Possono esserci una o più consegne indirette





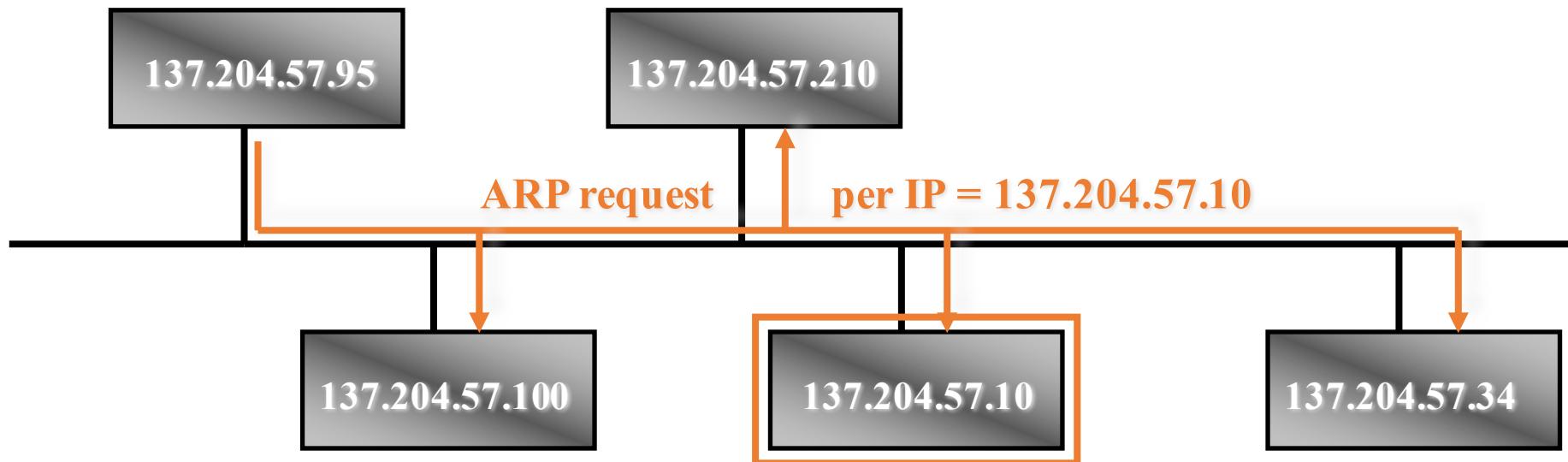
ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Address Resolution Protocol

# Relazione Indirizzi Fisici – Indirizzi IP

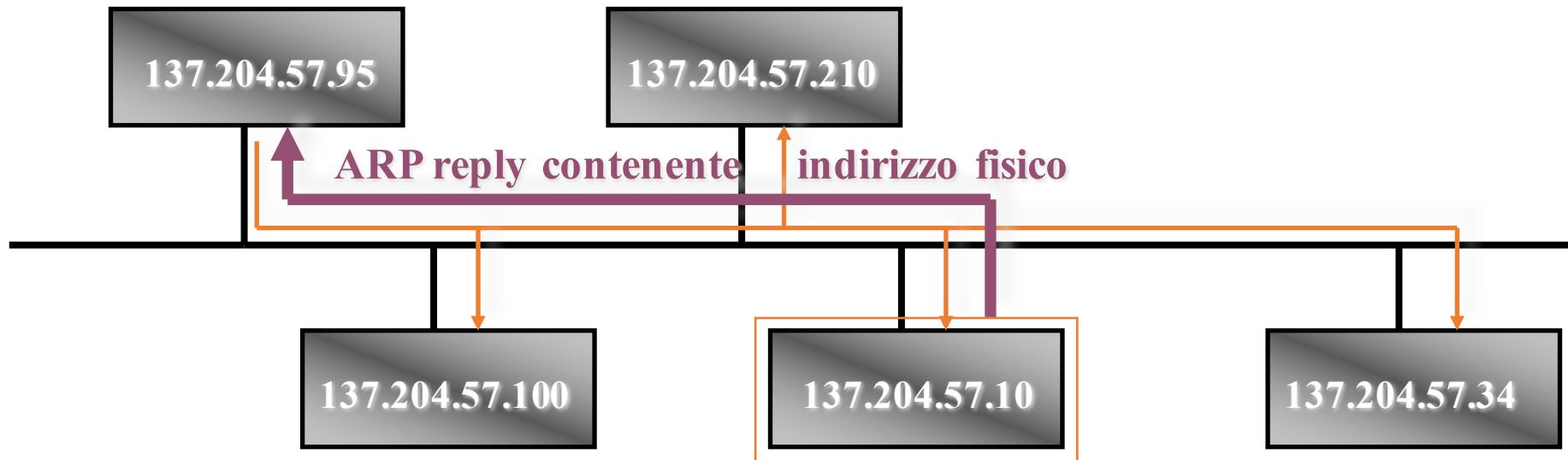
- Software di basso livello nasconde gli indirizzi fisici e consente ai livelli superiori di lavorare solo con indirizzi IP
- Gli host comunicano attraverso una **rete fisica** (ad es. LAN) quindi devono conoscere reciprocamente gli indirizzi fisici
- L' host A vuole mandare datagrammi a B, che si trova sulla stessa rete fisica e di cui conosce solo l' indirizzo IP
- Come si ricava l' indirizzo fisico di B dato il suo indirizzo IP?

# Address Resolution Protocol – ARP (RFC 826)



- Il nodo sorgente invia una trama broadcast (**ARP request**) contenente l' indirizzo IP del nodo destinazione
- Tutte le stazioni della rete locale leggono la trama broadcast

# Address Resolution Protocol - ARP (3)



- Il destinatario risponde al mittente, inviando un messaggio (**ARP reply**) che contiene il proprio indirizzo fisico
- Con questo messaggio host sorgente è in grado di associare l' appropriato indirizzo fisico all' IP destinazione
- Ogni host mantiene una tabella (**cache ARP**) con le corrispondenze fra indirizzi logici e fisici

# Comando ARP

**arp -a**

visualizza il contenuto della cache ARP con le diverse corrispondenze tra indirizzi IP e MAC

# Comando ARP – Esempio

```
Command Prompt

C:\>arp -a

Interface: 137.204.57.174 on Interface 0x10000003
Internet Address      Physical Address      Type
137.204.57.1           08-00-20-9c-9c-93    dynamic
137.204.57.88          00-60-b0-78-e8-fd    dynamic
137.204.57.180         00-10-4b-db-0a-3a    dynamic
137.204.57.181         00-30-c1-d5-ee-9b    dynamic
137.204.57.254         00-50-54-d9-ba-00    dynamic

C:\>ping -n 1 137.204.57.177

Pinging 137.204.57.177 with 32 bytes of data:

Reply from 137.204.57.177: bytes=32 time<10ms TTL=128

Ping statistics for 137.204.57.177:
  Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>arp -a

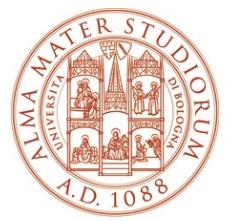
Interface: 137.204.57.174 on Interface 0x10000003
Internet Address      Physical Address      Type
137.204.57.1           08-00-20-9c-9c-93    dynamic
137.204.57.177         00-b0-d0-ec-46-62    dynamic
137.204.57.180         00-10-4b-db-0a-3a    dynamic
137.204.57.181         00-30-c1-d5-ee-9b    dynamic
137.204.57.254         00-50-54-d9-ba-00    dynamic

C:\>_
```



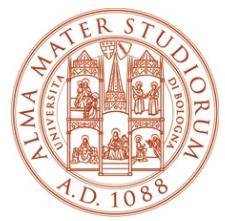
ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Classless VS Classfull la logica degli indirizzi IP



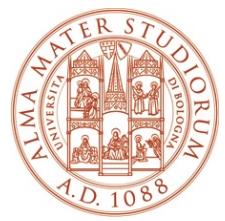
# IP e netmask

- Il numero IP ha valore assoluto in rete
  - Un numero IP pubblico deve essere unico su Internet
  - I numeri IP sorgente e destinazione caratterizzano il datagramma in quanto parte della sua intestazione
- La netmask è relativa al singolo nodo
  - Non viene trasportata nell'intestazione del datagramma
  - È parte della tabella di routing dei singoli nodi
  - Ai medesimi indirizzi possono corrispondere netmask diverse in nodi diversi (route aggregation)
- È sempre stato così?
  - NO: inizialmente la suddivisione net-ID e host-ID era assoluta

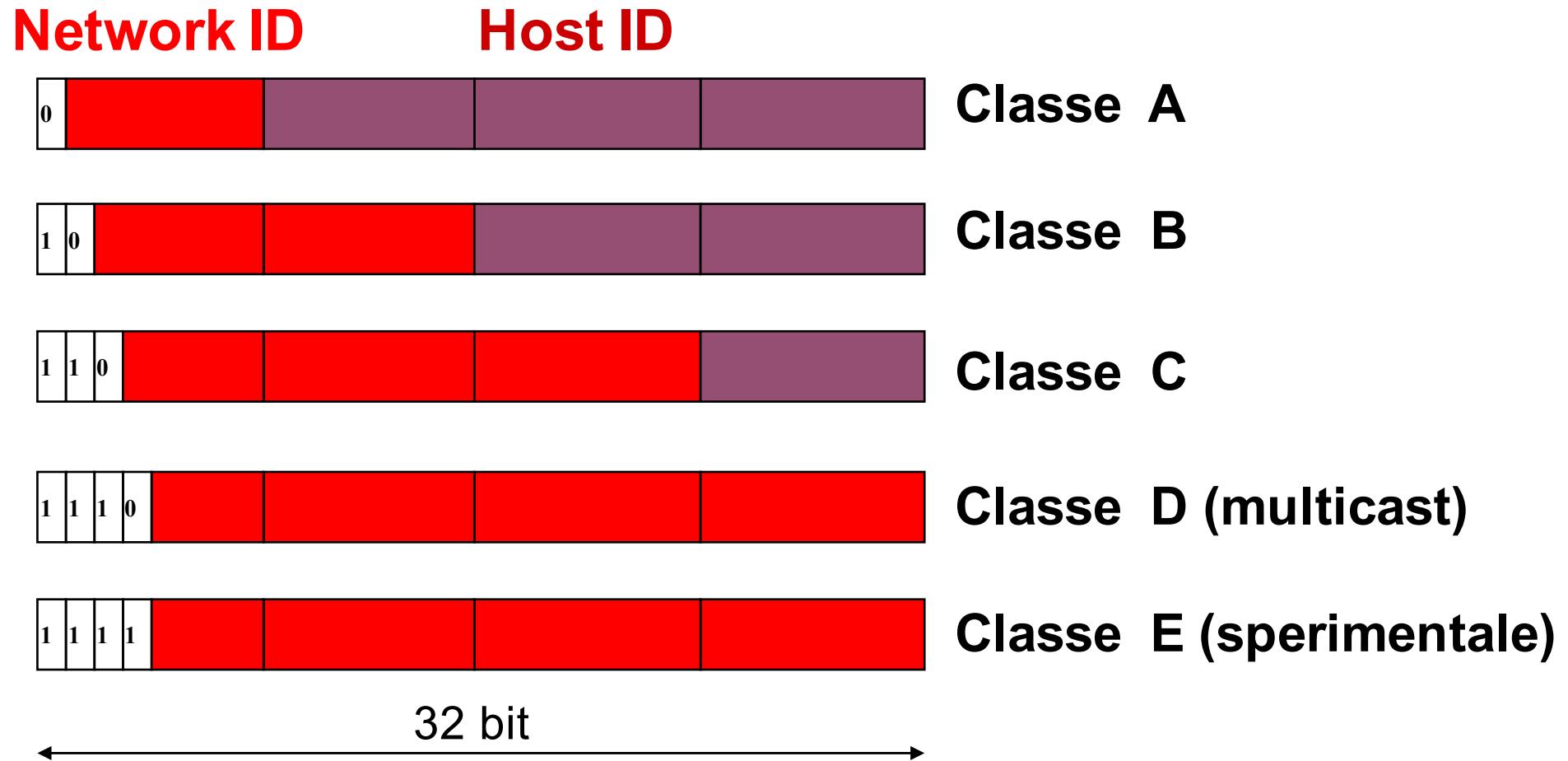


# Classe delle reti

- Durante la fase iniziale di Internet furono definite diverse “**classi**” di network differenziate per **dimensione**
  - La parte iniziale del Net-ID differenzia le classi
    - 0 classe A
    - 10 classe B
    - 110 classe C
  - La definizione delle classi è standard e quindi nota a tutti
  - I router riconoscono la classe di una rete dai primi bit dell’ indirizzo
    - Ricavano di conseguenza il Net-ID

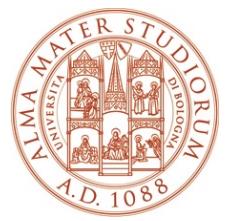


# Classi di indirizzi



**Network ID :** identifica una rete IP

**Host ID :** identifica i singoli calcolatori della rete



# Intervalli di indirizzi

- Classe A: da 0.0.0.0 a 127.255.255.255
- Classe B: da 128.0.0.0 a 191.255.255.255
- Classe C: da 192.0.0.0 a 223.255.255.255
- Classe D: da 224.0.0.0 a 239.255.255.255
- Classe E: da 240.0.0.0 a 255.255.255.255
  
- Indirizzi riservati (RFC 1700)
  - 0.0.0.0 indica l'host corrente senza specificarne l'indirizzo
  - Host-ID **tutto a 0** viene usato per **indicare la rete**
  - Host-ID **tutto a 1** è l'indirizzo di **broadcast** per quella rete
  - 0.x.y.z indica un certo Host-ID sulla rete corrente senza specificare il Net-ID
  - 255.255.255.255 è l'indirizzo di broadcast su Internet
  - 127.x.y.z è il **loopback**, che redirige i datagrammi agli strati



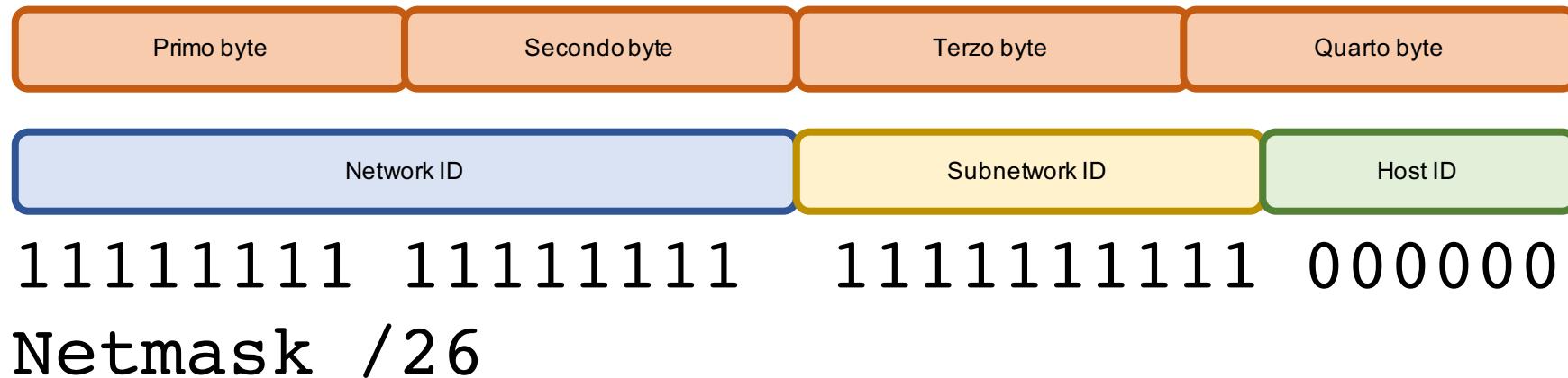
# Le sottoreti

- A un'amministrazione è assegnata una network
  - L' amministrazione potrebbe essere suddivisa in sotto-amministrazioni *logicamente separate*
  - Converrebbe “*frammentare*” la network in “**sub-network**” da assegnare alle sotto-amministrazioni
- Si decide localmente una sotto-ripartizione Net/Host ID **indipendente dalle classi**
- Si frammenta l' Host-ID in due parti:
  - la prima identifica la sottorete (**subnet-ID**)
  - la seconda identifica i singoli host della sottorete
- La ripartizione deve essere *locale* e *reversibile*
  - Tutta Internet vede comunque una certa network come un' entità unitaria



# Subnetting

- La suddivisione è locale alla singola interfaccia
  - Deve essere configurabile localmente
- Si personalizza la **Netmask**

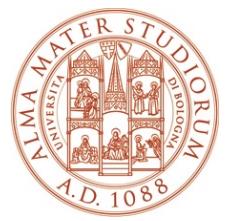




# Esempio: Università di Bologna

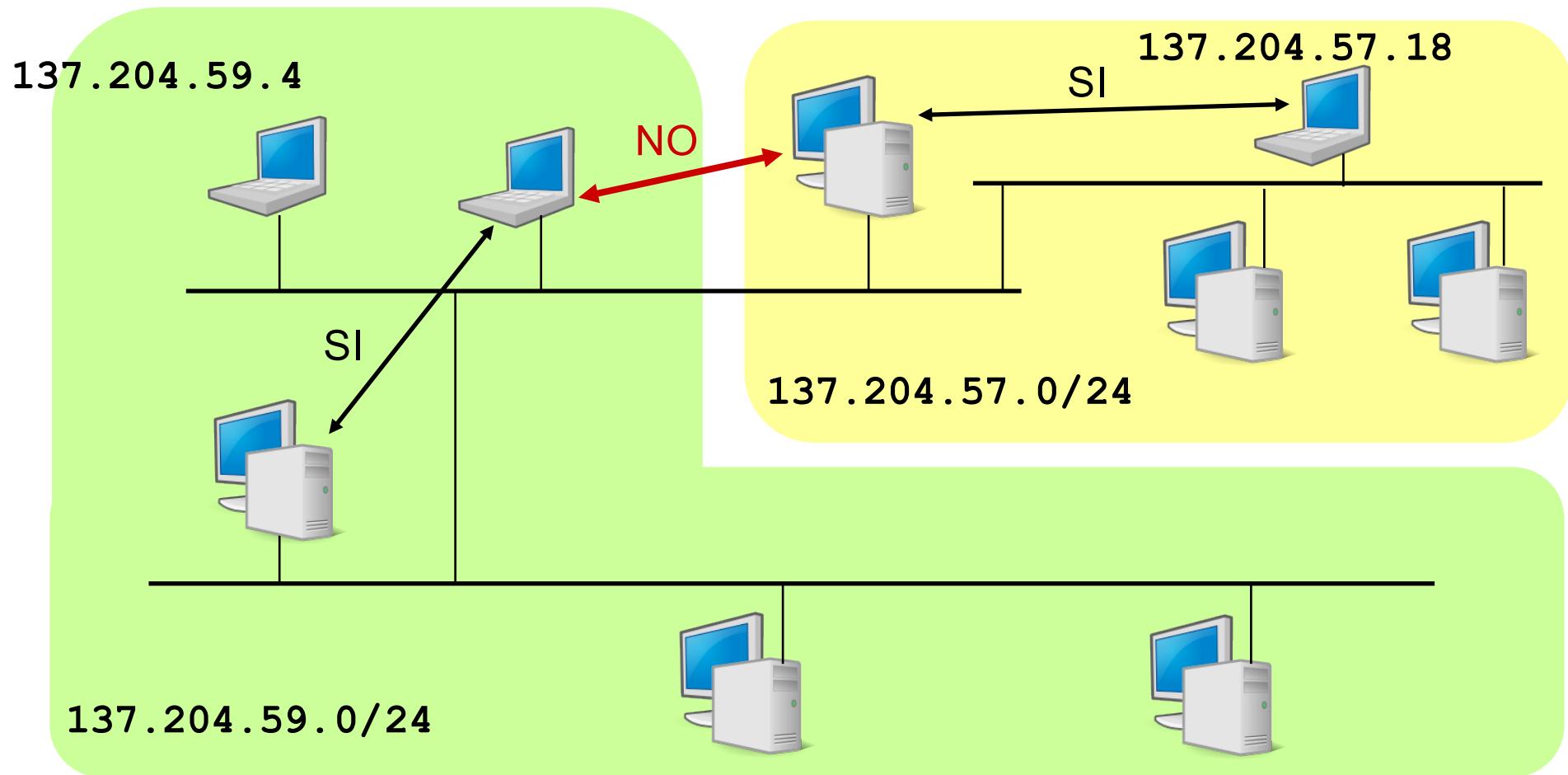
- Una network di classe B (137.204.0.0)
  - Numerose entità distinte nella stessa amministrazione
    - Facoltà, Dipartimenti, Centri di ricerca ecc.
  - Si suddivide la rete (network) in sottoreti (subnetwork)
- Il primo byte del Host-ID viene utilizzato come indirizzo di sottorete
  - Dalla network di classe B si ricavano 254 network della dimensione di una classe C

**Netmask = 255.255.255.0**



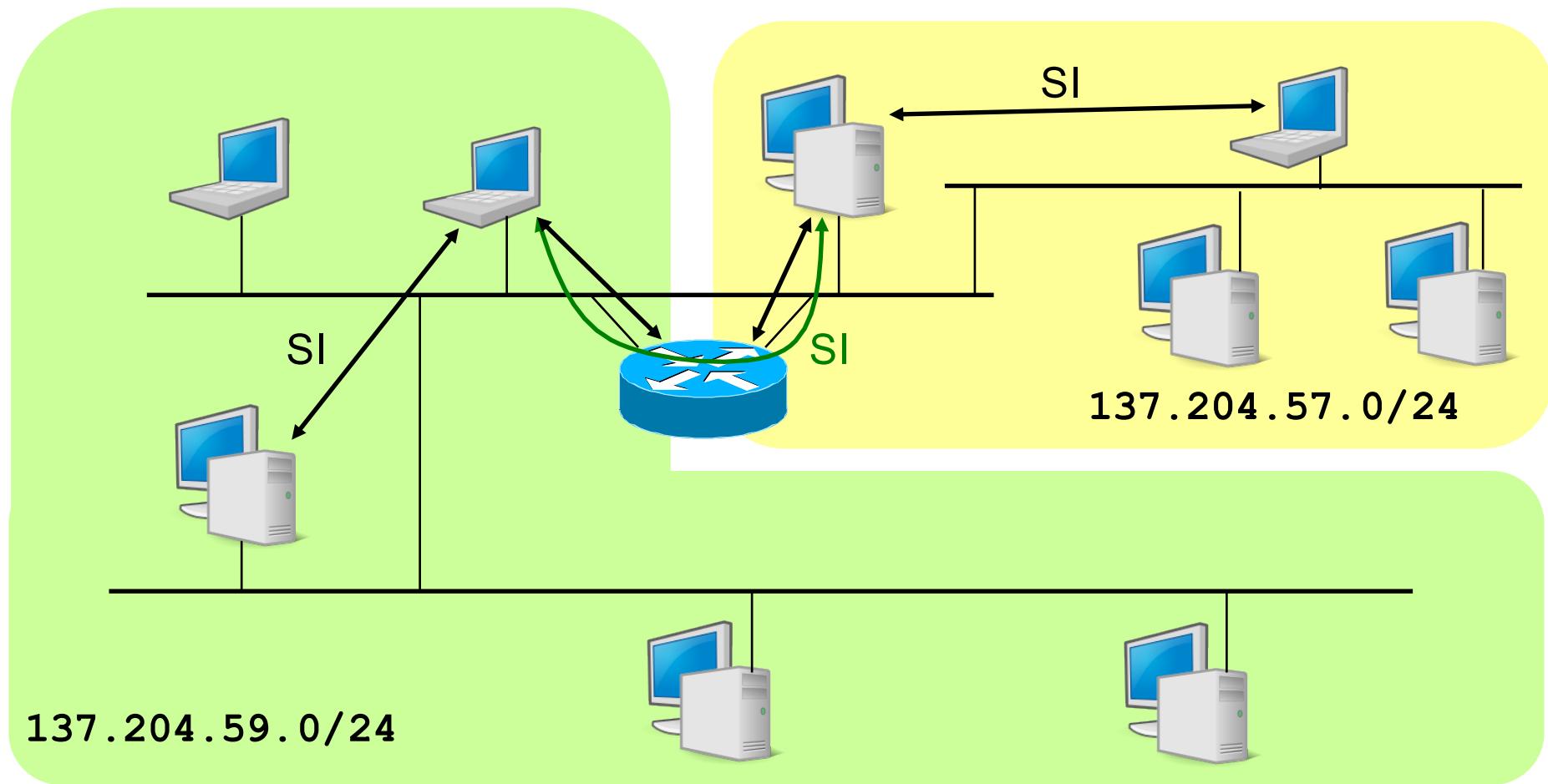
# Subnetting

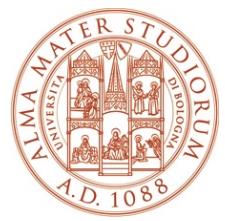
- Subnet diverse sono di fatto Network diverse e quindi non comunicano
- È necessario un gateway



# Subnetting

- Il Gateway permette instradamento indiretto fra le Subnetwork





# CIDR

- Con la grande diffusione di Internet la rigida suddivisione nelle 3 classi rendono l' instradamento poco flessibile e scalabile
- **CIDR** (RFC 1519) Classless InterDomain Routing
  - Si decide di rompere la logica delle classi nei router
  - La dimensione del Net-ID può essere qualunque
  - Le tabelle di routing devono **comprendere anche le Netmask**
  - Generalizzazione del subnetting/supernetting
    - reti IP definite da **Net-ID/Netmask**



# Obiettivi del CIDR

- Allocazione di reti IP di dimensioni variabili
  - utilizzo più efficiente dello spazio degli indirizzi
- Accorpamento delle informazioni di routing
  - più reti contigue rappresentate da un' unica riga nelle tabelle di routing
- Miglioramento di due situazioni critiche
  - Limitatezza di reti di classe A e B
  - Crescita esplosiva delle dimensioni delle tabelle di routing



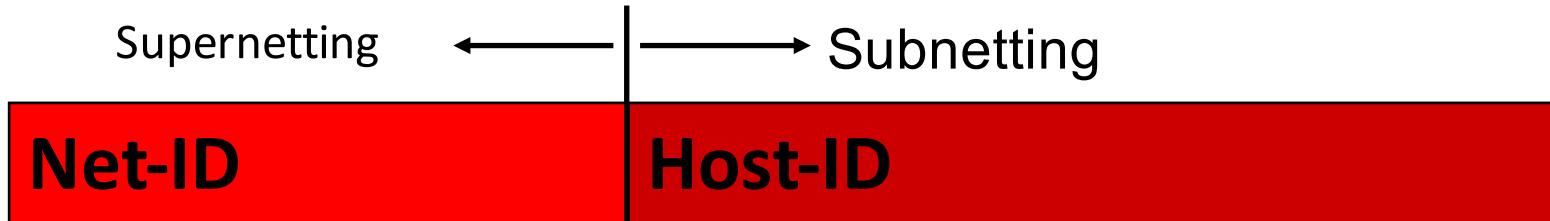
# Supernetting

- Raggruppare più reti con indirizzi consecutivi
  - Indicarle nelle tabelle di routing con una sola entry accompagnata dalla opportuna Netmask
- Es. Un ente ha bisogno di circa 2000 indirizzi IP
  - una rete di classe B è troppo grande (64K indirizzi)
  - meglio 8 reti di classe C ( $8 \times 256 = 2048$  indirizzi) dalla 194.24.0.0 alla 194.24.7.0
- **Supernetting:** si accorpano le 8 reti contigue in un'unica super-rete:
  - Identificativo: 194.24.0.0/21
  - Supernet mask: 255.255.248.0
  - Indirizzi: 194.24.0.1 – 194.24.7.254
  - Broadcast: 194.24.7.255

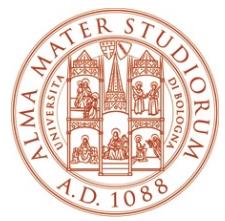


# Supernetting

- Subnetting e Supernetting sono operazioni duali
  - Subnetting → **n** bit del Host-ID diventano parte del Net-ID
  - Supernetting → **n** bit del Net-ID diventano parte dell' Host-ID

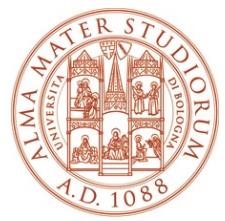


- Accorpamento di **N** reti IP (**N = 2<sup>n</sup>**)
  - **contigue**:
    - $194.24.0.0/24 + 194.24.1.0/24 = 194.24.0.0/23$
    - $194.24.0.0/24 + 194.24.2.0/24$  = non contigue
  - **allineate** secondo i multipli di  $2^n$ 
    - $194.24.0.0/24 + .1.0/24 + .2.0/24 + .3.0/24 = 194.24.0.0/22$
    - $194.24.2.0/24 + .3.0/24 + .4.0/24 + .5.0/24$  = non allineate

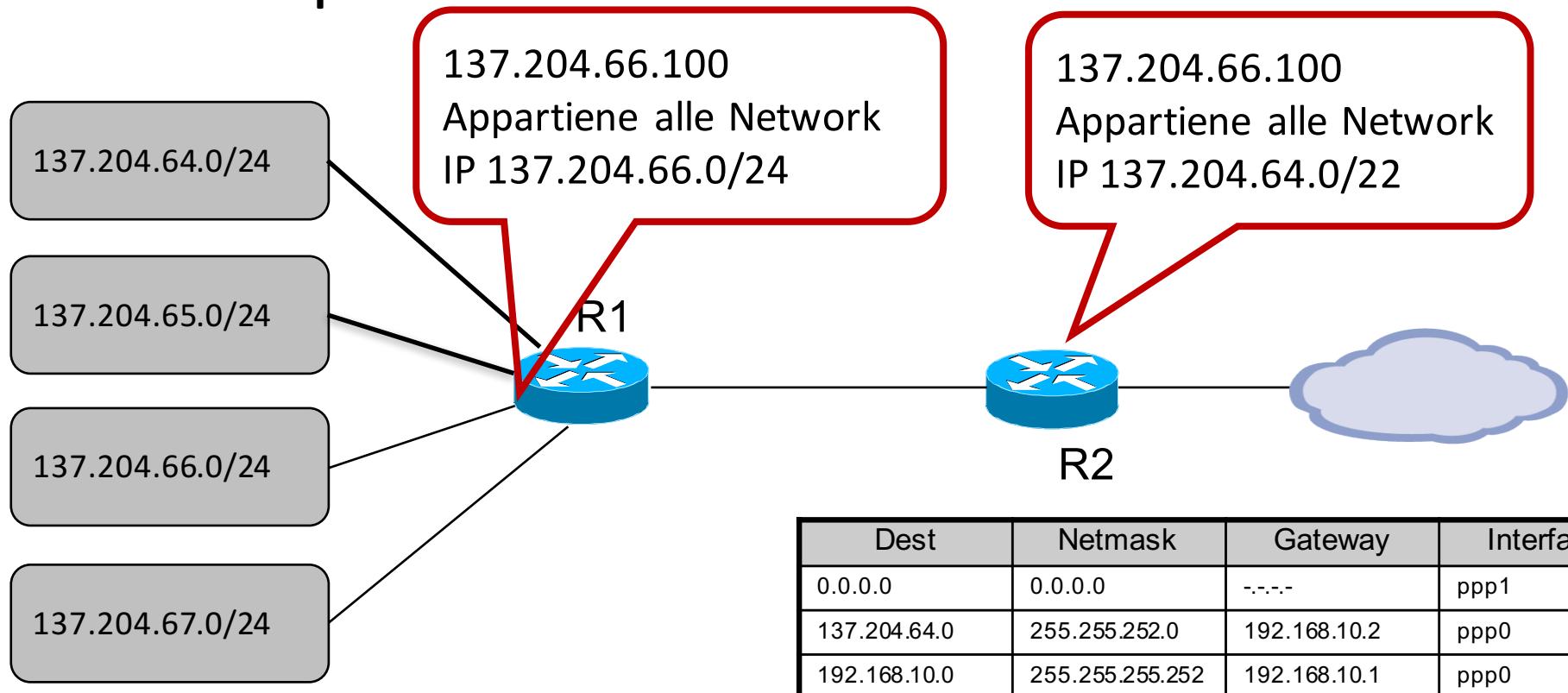


# Oggi

- La distinzione fra Net-ID e Host-ID è locale funzione della Netmask
- Lo stesso indirizzo può essere interpretato in modo diverso in punti diversi della rete
- Tutte le tabelle di instradamento devo contenere la colonna delle Netmask



# Esempio



| Dest         | Netmask         | Gateway        | Interface |
|--------------|-----------------|----------------|-----------|
| 0.0.0.0      | 0.0.0.0         | 192.168.10.1   | ppp0      |
| 137.204.64.0 | 255.255.255.0   | 137.204.64.254 | en0       |
| 137.204.65.0 | 255.255.255.0   | 137.204.65.254 | en1       |
| 137.204.66.0 | 255.255.255.0   | 137.204.66.254 | en2       |
| 137.204.67.0 | 255.255.255.0   | 137.204.67.254 | en3       |
| 192.168.10.0 | 255.255.255.252 | 192.168.10.2   | ppp0      |