PARTE 4c

LIVELLO IP
(La “dorsale” di Internet)

Modulo 9:
Routing su scala geografica

Routing gerarchico

- **Nella realtà, i router di Internet:**
  - non rappresentano un insieme omogeneo di risorse
  - non eseguono lo stesso algoritmo di routing
- **Diversi motivi:**
  - Scalabilità: all'aumentare del numero di router, l'overhead degli algoritmi di routing diviene proibitivo

→ occorre ridurre la complessità del calcolo del cammino

  - Autonomia amministrativa: un'organizzazione dovrebbe/vorrebbe scegliere autonomamente come amministrare il traffico tra i propri router

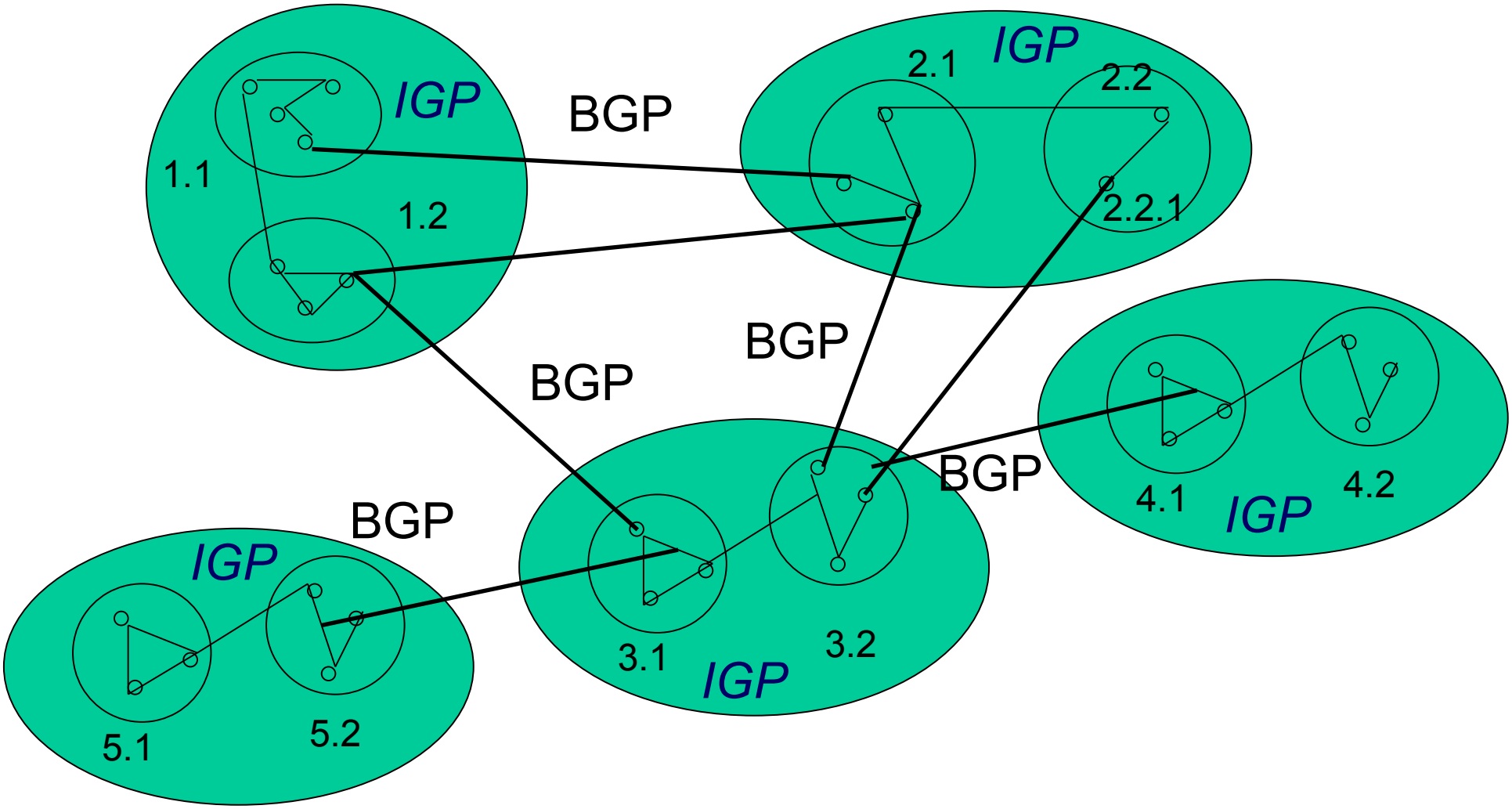
Autonomous Systems per il routing

- **I router vengono aggregati in “regioni” o sistemi autonomi (AS)**
- **Un AS è un insieme di nodi e router gestiti da un’unica entità di controllo centrale (es., stesso ISP)**
- **Ciascun AS ha un numero identificativo assegnato da una authority di registrazione Internet:**
  - Il numero è compreso fra 1 e 64511
  - I numeri di AS compresi nell’intervallo 64512-65535 sono riservati

Autonomous Systems per il routing (2)

- **Per il routing all'interno di un AS (routing Intra-AS) i router utilizzano qualche Interior Gateway Protocol dove i router di un AS possono possedere un'informazione completa su tutti gli altri router dell'AS**
- **Per il routing verso altri AS (routing Inter-AS) viene utilizzato qualche Exterior Gateway Protocol (prima EGP, oggi BGP)**
- **Ciascun AS può usare metriche multiple per il routing interno, ma appare come un unico AS ad altri AS**

Esempio con 5 AS

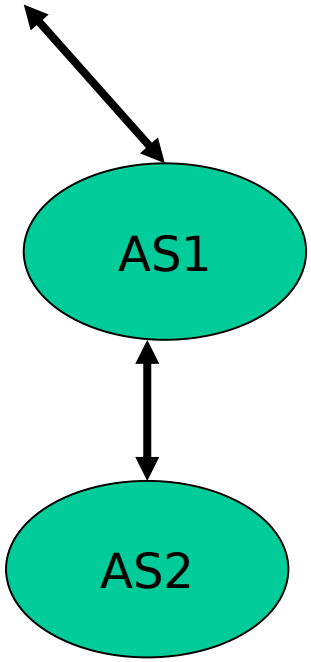


Politiche di routing negli AS

- **Le politiche di routing sono le regole per decidere come instradare il traffico**
  - Se sono un AS, quale traffico accetto di far passare per la mia rete?
  - Ci sono spesso accordi commerciali alla base di queste decisioni (RICORDARE: peering point e IX)

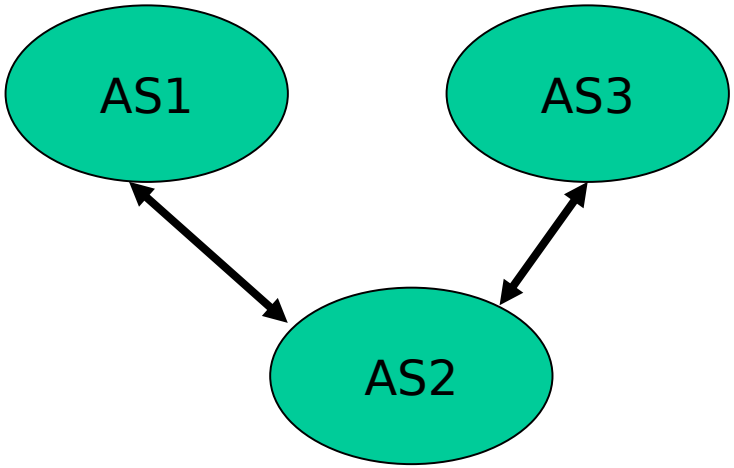
- **Ogni AS vuole poter decidere le proprie politiche e potrebbe anche non volerle fare conoscere agli altri AS**

Tipi di AS



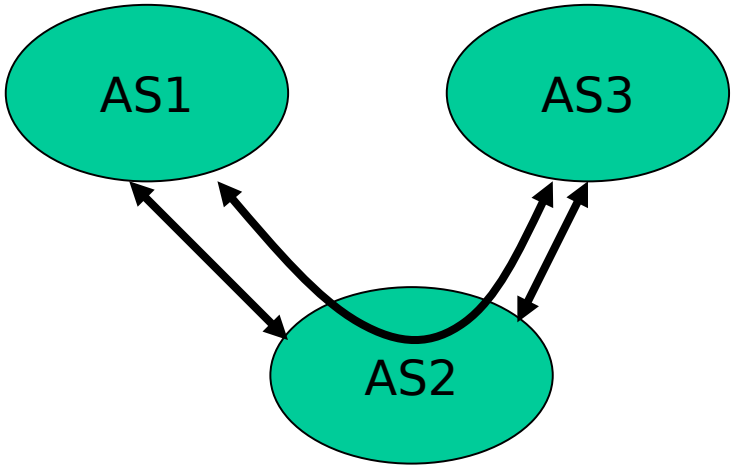
AS2: Stub

AS con una sola connessione ad un altro AS



AS2: Multi-homed

AS connesso con diversi AS, ma non permette traffico che non è generato o diretto verso l'AS (accetta solo traffico locale)



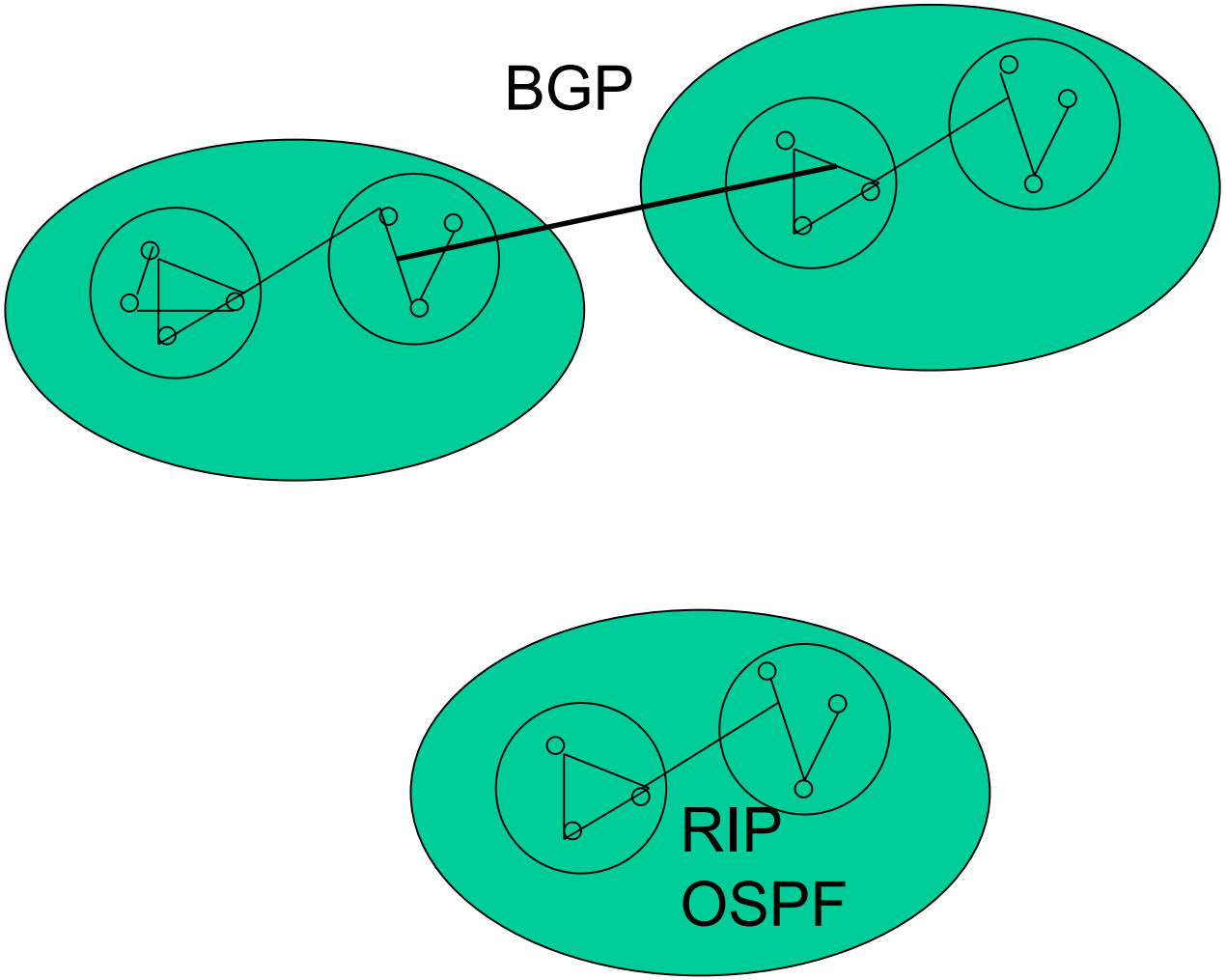
AS2: Transit

AS connesso a diversi AS che consente di fare da tramite per gli AS a cui è collegato (accetta sia traffico locale sia traffico in transito)

Protocolli di routing: inter-AS, intra-AS

- **Principale protocollo di routing inter-AS**
  - Border Gateway Protocol (BGP)
    - Algoritmo di routing distribuito
    - E’ oggi lo standard de facto per il routing tra AS
- **Principali protocolli di routing intra-AS**
  - Routing Information Protocol (RIP)
    - Algoritmo di routing distribuito (Distance vector protocol)
  - Open Shortest Path First (OSPF)
    - Algoritmo di routing centralizzato (Link state protocol)
    - Successore di RIP

Protocolli di routing: inter-AS, intra-AS



Modulo 10: Protocollo BGP (Inter-AS)

Un po' di storia…

- **Fino agli anni '80: EGP**
  - Storicamente il primo protocollo ad essere usato per il routing inter-AS
  - Presuppone una rete con topologia ad albero senza cicli (come la vecchia ARPANET)
  - Limiti nella massima dimensione delle reti gestibili
  - Entra in crisi con l'introduzione delle dorsali Internet e dei cammini multipli tra nodi
- **BGP viene introdotto per sostituire EGP**
- **BGP usa un algoritmo distribuito di tipo "distance vector"**

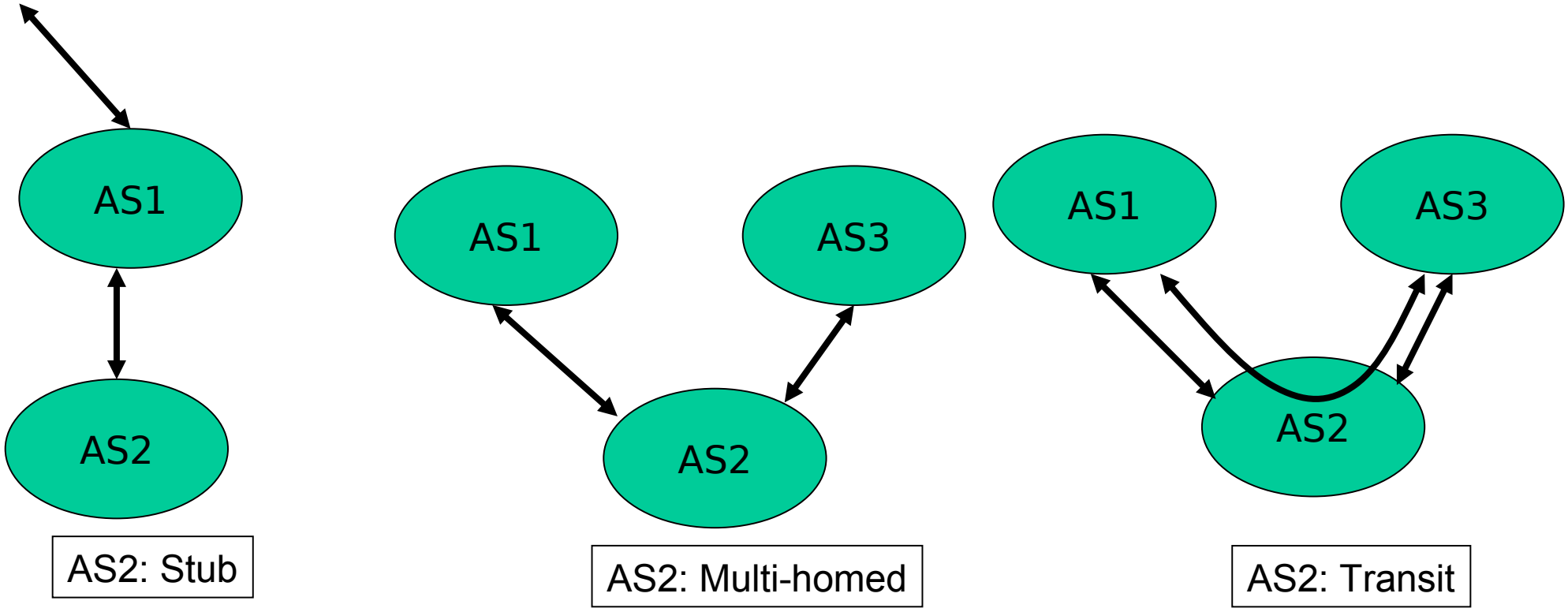
BGP

- **Border Gateway Protocol (versione 4): RFC 1771**
- **E’ un protocollo complesso, ma fondamentale per il funzionamento di Internet, in quanto è il protocollo delle dorsali Internet per muoversi da un AS ad un altro in modo completamente decentralizzato**
- **Utilizzato dagli ISP e non dai piccoli-medi utenti**
- **Può essere utilizzato anche come protocollo intra-AS nel caso di AS molto grandi (in quanto il protocollo intra-AS OSPF non scala molto bene)**

BGP e tipi di AS

NON NECESSARIO

NECESSARIO

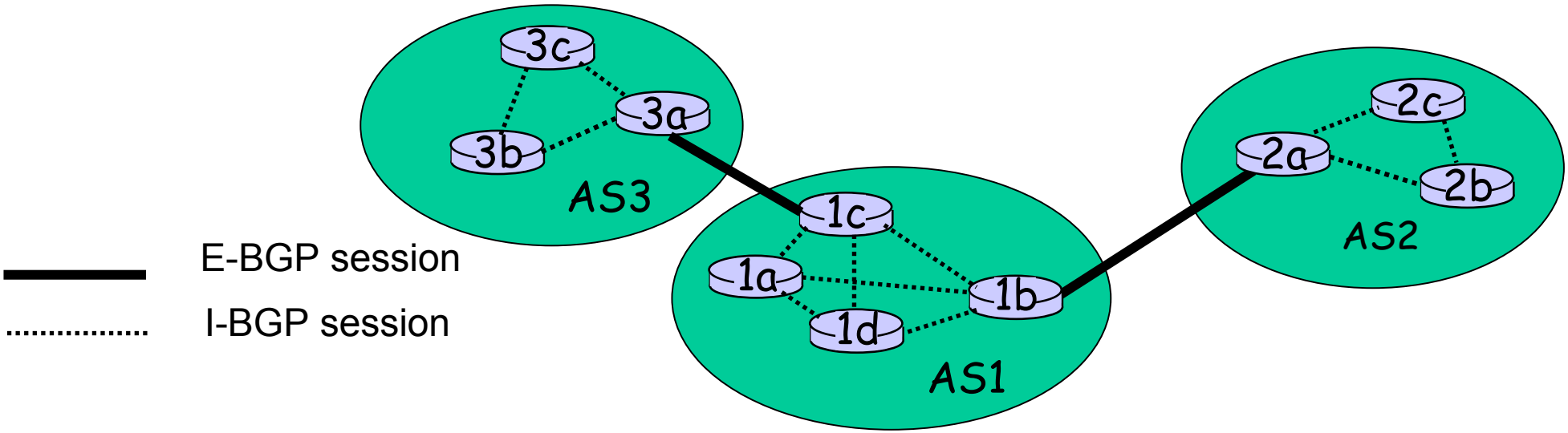


Funzioni BGP e prefissi

- **Funzioni principali**
  - Scambiare informazioni di raggiungibilità tra AS confinanti, detti peer (configurando manualmente i router)
  - Propagare le informazioni di raggiungibilità a tutti i router all'interno di un AS → meccanismo distribuito basato sull'algoritmo Path Vector (della classe Distance Vector Protocol)
  - Determinare i percorsi migliori in base a informazioni di raggiungibilità e policy di routing (non solo metriche!)
- **Le destinazioni sono indicate con prefissi che rappresentano una o più sottoreti (ampio utilizzo di aggregazione di indirizzi per ridurre le entry della routing table)**

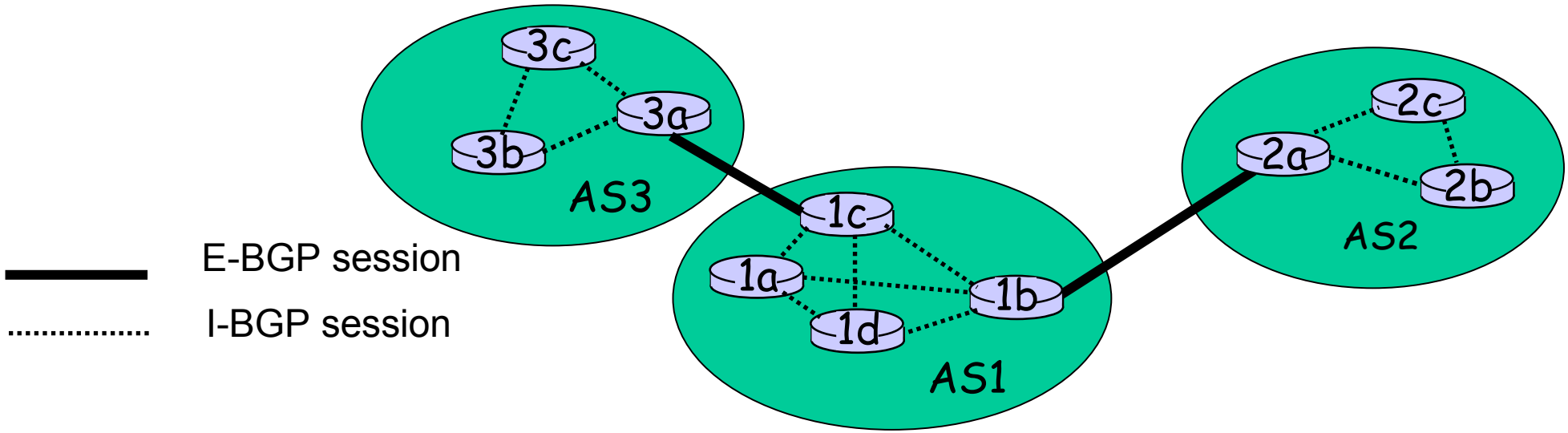
Sessioni BGP

- **BGP fa uso di connessioni TCP [si vedrà in seguito] semi-permanenti per far comunicare i router confinanti (BGP peers)**
- **I due peer BGP che si scambiano messaggi sulla connessione TCP formano una sessione BGP**



Sessioni BGP

- **Sessione esterna (E-BGP) tra router di AS diversi**
- **Sessione interna (I-BGP) tra router dello stesso AS**

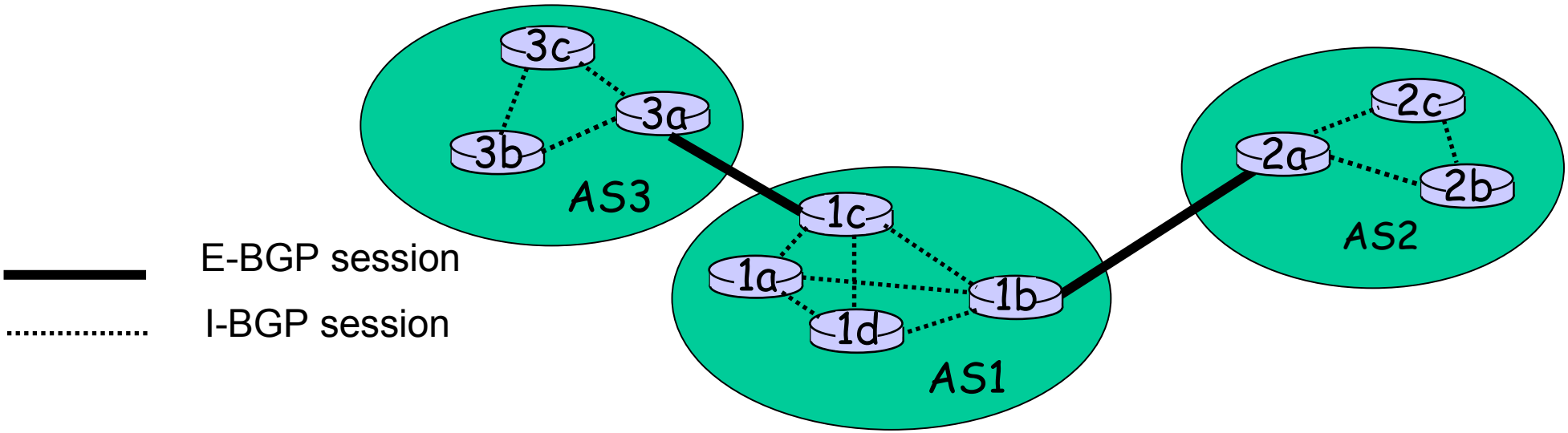


Router e BGP

- **Transit router: router che gestiscono traffico I-BGP all'interno dell'AS**
- **I transit router devono essere configurati a maglia (mesh), cioè tutti devono essere peer di tutti gli altri. Questo pone dei problemi di scalabilità, risolti mediante confederazioni**
- **Border router (o edge router) router che gestiscono traffico E-BGP tra diversi AS**
- **La scelta dei peer di un border router dipende dalle policy del gestore dell'AS**

Distribuzione di informazioni per la raggiungibilità

- **3a annuncia a 1c i prefissi di rete raggiungibili da AS3 attraverso una sessione E-BGP**
- **1c usa I-BGP per distribuire le informazioni di raggiungibilità a tutti i router in AS1 (1a – 1d – 1b)**
- **1b annuncia a 2a i prefissi raggiungibili da AS3 e AS1 attraverso una sessione E-BGP**
- **Quando un router viene a conoscenza di un nuovo prefisso, crea una nuova riga nella propria tabella di routing**



Prefissi e attributi BGP

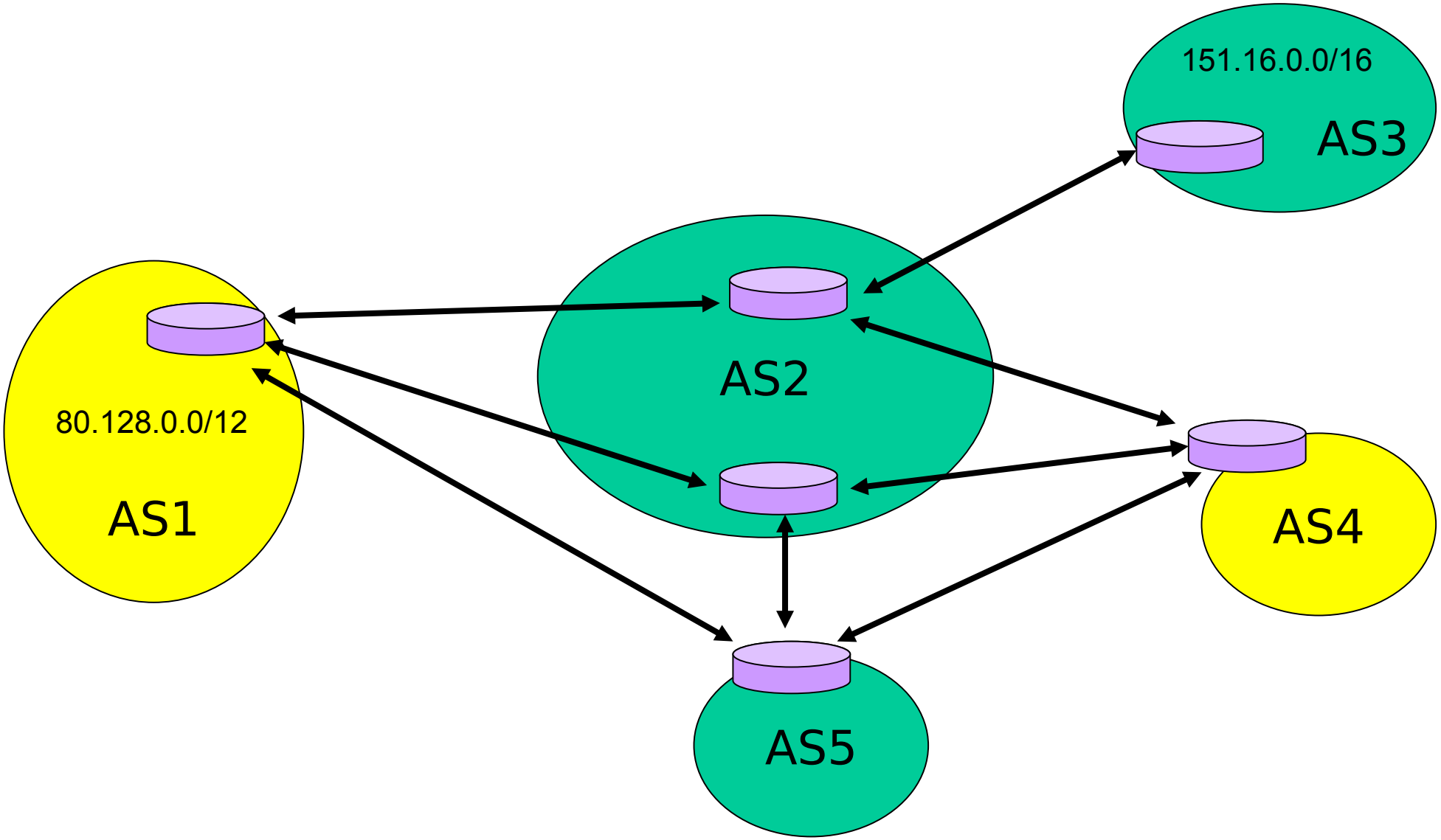
- **“Annunciare un prefisso” equivale alla “promessa” di inoltrare i pacchetti su un percorso verso il prefisso di destinazione**
- **L’annuncio di un prefisso comprende anche attributi BGP, tra cui i più importanti sono:**
  - AS-PATH: elenco degli AS attraverso cui è passato l’annuncio del prefisso (strada attraversata)
    NOTA: AS-PATH da una misura della lunghezza del cammino, può essere usata per far prefereire un cammino ad un'altro
  - NEXT-HOP: lo specifico router nel next-hop AS da cui giunge l’annuncio (possibili più collegamenti tra gli AS)

Selezione del percorso

- **Un router può venire a conoscenza di più di un percorso verso un prefisso**

  → deve selezionarne uno
- **Quando un border router riceve un annuncio utilizza le policy locali per decidere se accettare/scartare l'annuncio**
- **Varie possibili regole:**
  - Valore di preferenza locale: policy
  - AS-PATH più breve
  - Router di NEXT-HOP più vicino
  - Altri criteri

Path Vector: esempio path alternativi tra AS1 e AS4



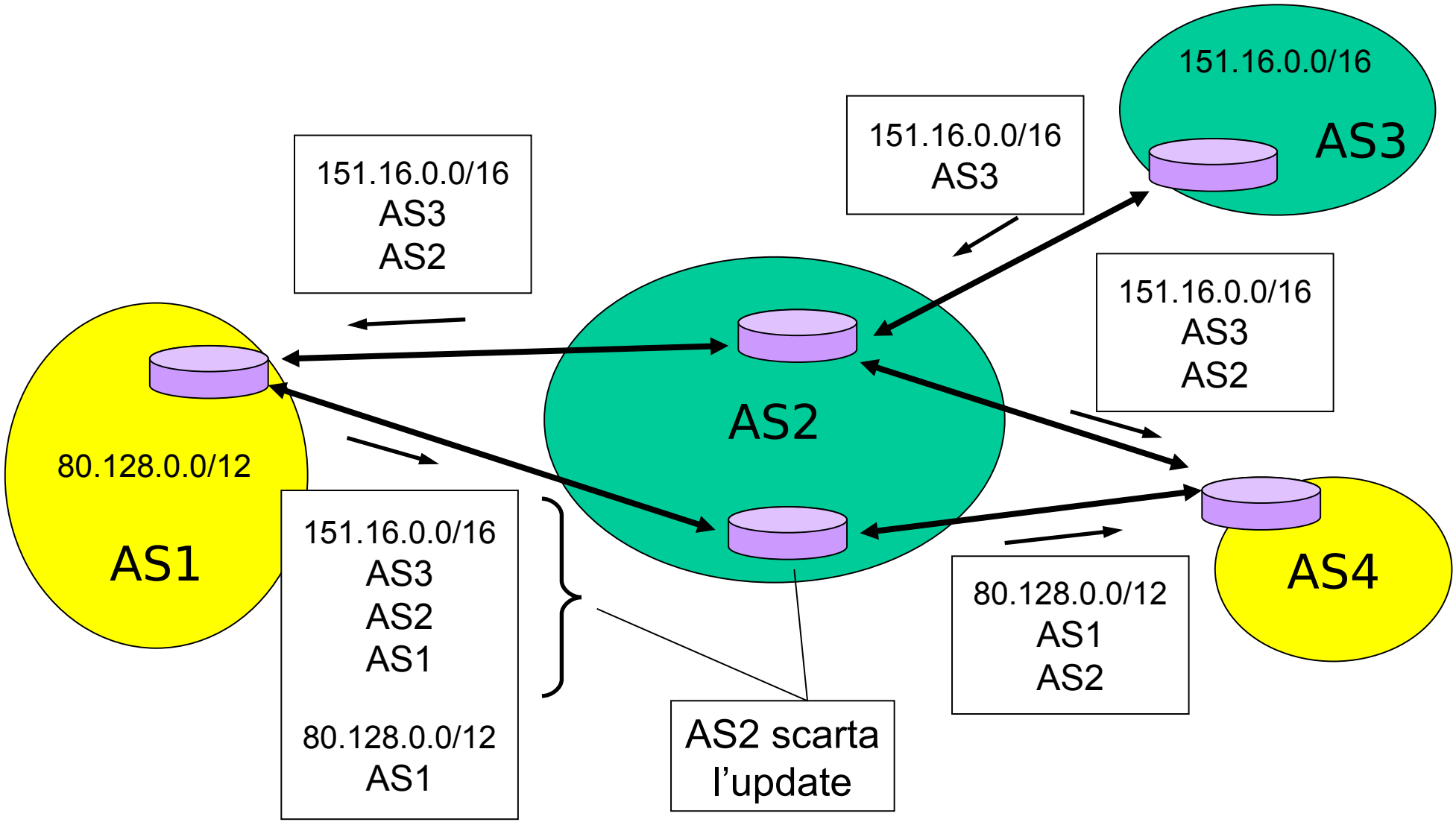
Algoritmo Path Vector

- **BGP usa Path Vector, una variante dell'algoritmo distribuito Distance Vector Protocol**
- **Ogni routing update contiene non solo informazioni adiacenti, ma sull'intero cammino verso la destinazione attraverso gli AS**

Algoritmo Path Vector

- **Individuazione dei loop → risolve problema degli algoritmi distance vector**
  - Quando un AS riceve un update riguardo un percorso, controlla se il percorso contiene se stesso come info
    - Se sì, scarta l'update
    - Se no, aggiunge se stesso e (eventualmente) propaga l'update
- **Vantaggi:**
  - Uso di metriche e policy locali nelle decisioni
  - Il protocollo garantisce che non ci siano loop

Path Vector: esempio



Modulo 11: Protocollo OSPF (Intra-AS)

Open Short Path First

- **“open” = disponibile pubblicamente (a differenza del protocollo EIGRP della Cisco)**
- **E’ un link state protocol**
- **Ha varie funzioni migliorative rispetto a RIP**

- **Attualmente, si tende a utilizzare OSPF all’interno di AS di primo livello**

Open Short Path First

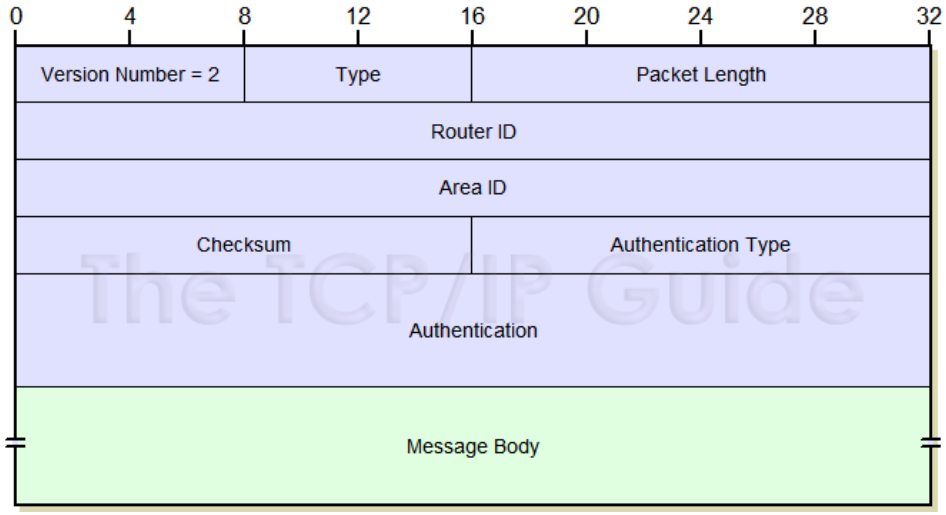
- **Il routing si basa sull'algoritmo centralizzato Link State**
  - Topologia e costi noti ad ogni nodo
  - Si calcola l'albero dei cammini di costo minimo mediante l'algoritmo di Dijkstra
  - Si memorizza tale albero nel cosiddetto "link state database" che viene distribuito a tutti i router
- **Gli aggiornamenti sono inviati in broadcast ai router dell'intero AS (flooding)**
  - I messaggi OSPF viaggiano direttamente su IP
  - Il "link state database" viene inviato periodicamente (almeno ogni 30 minuti) anche se non è cambiato

Caratteristiche di OSPF (non in RIP)

- **Sicurezza: autenticazione dei messaggi OSPF con algoritmi di crittografia**
- **Percorsi multipli con costo uguale: possibilità di usare più percorsi per instradare il traffico (mentre è solo uno in RIP)**
- **Supporto integrato per instradamento unicast e multicast: multicast OPSF (MOSPF) usa lo stesso database di collegamenti usato da OSPF**
- **Struttura gerarchica degli AS: possibilità di strutturare grandi domini di instradamento in gerarchie di AS**

Struttura messaggi OSPF (v2)

- **Header + Corpo**
  - Header comune
  - Corpo dipende dal tipo
- **Numero versione**
- **Tipo**
  - 1 Hello
  - 2 DB description
  - 3 Link state req
  - 4 link state update
  - 5 Link state ACK



- **Dimensione (in bytes, incluso header)**
- **Router ID (tipicamente IP dell'interfaccia da cui il messaggio è spedito)**
- **Area cui il messaggio appartiene. Usato quando OSPF divide il suo dominio in aree**
- **Checksum (comprende tutto tranne i campi di autenticazione)**
- **Tipo di autenticazione**
  - 0 Nessuna autenticazione
  - 1 Password
  - 2 Autenticazione crittografica
- **Informazioni per autenticazione**

Significato dei messaggi OSPF

- **Hello**
  - Scoperta di router adiacenti
  - Scambio di parametri sul funzionamento OSPF
  - Pacchetti Hello tipicamente mandati a indirizzo multicast 224.0.0.5
- **DB description**
  - Contenuto del Link State DB
- **Link State Request**
  - Richiesta di informazioni relative a una porzione del Link State DB

Significato dei messaggi OSPF

- **Link State Update**
  - Informazioni relative a link
  - Mandato in risposta a request
  - Usato per propagare periodicamente gli update nella struttura della rete
- **Link State ACK**
  - Acknowledgement della ricezione di un messaggio di update

Analisi di un pacchetto OSPF

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
2	9.999097	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
3	19.999284	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
4	30.000172	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
5	39.999479	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
6	50.001667	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
7	60.000453	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet
8	69.999831	155.185.179.254	224.0.0.5	OSPF	98	Hello Packet

```
▸ Frame 3: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
▸ Ethernet II, Src: HewlettP_08:66:00 (00:0e:7f:08:66:00), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
▸ 802.1Q Virtual LAN, PRI: 6, CFI: 0, ID: 1450
▸ Internet Protocol Version 4, Src: 155.185.179.254, Dst: 224.0.0.5
▾ Open Shortest Path First
  ▾ OSPF Header
      Version: 2
      Message Type: Hello Packet (1)
      Packet Length: 44
      Source OSPF Router: 155.185.178.0
      Area ID: 155.185.178.0
      Checksum: 0x0000 (None)
      Auth Type: Cryptographic (2)
      Auth Crypt Key id: 1
      Auth Crypt Data Length: 16
      Auth Crypt Sequence Number: 5944763
      Auth Crypt Data: 147dba79b7f23d602cc8ac4870db98a3
  ▾ OSPF Hello Packet
      Network Mask: 255.255.254.0
      Hello Interval [sec]: 10
    ▾ Options: 0x02 ((E) External Routing)
        0... .... = DN: Not set
        .0.. .... = O: Not set
        ..0. .... = (DC) Demand Circuits: Not supported
        ...0 .... = (L) LLS Data block: Not Present
        .... 0... = (N) NSSA: Not supported
        .... .0.. = (MC) Multicast: Not capable
        .... ..1. = (E) External Routing: Capable
        .... ...0 = (MT) Multi-Topology Routing: No
      Router Priority: 1
      Router Dead Interval [sec]: 40
      Designated Router: 155.185.179.254
      Backup Designated Router: 0.0.0.0
```

```
0000  01 00 5e 00 00 05 00 0e  7f 08 66 00 81 00 c5 aa   ..^..... ..f.....
0010  08 00 45 c0 00 50 fe 80  00 00 01 59 8a 57 9b b9   ..E..P.. ...Y.W..
0020  b3 fe e0 00 00 05 02 01  00 2c 9b b9 b2 00 9b b9   ........ .,......
0030  b2 00 00 00 00 02 00 00  01 10 00 5a b5 bb ff ff   ........ ...Z....
0040  fe 00 00 0a 02 01 00 00  00 28 9b b9 b3 fe 00 00   ........ .(......
0050  00 00 14 7d ba 79 b7 f2  3d 60 2c c8 ac 48 70 db   ...}.y.. =`,..Hp.
0060  98 a3                                             ..
```

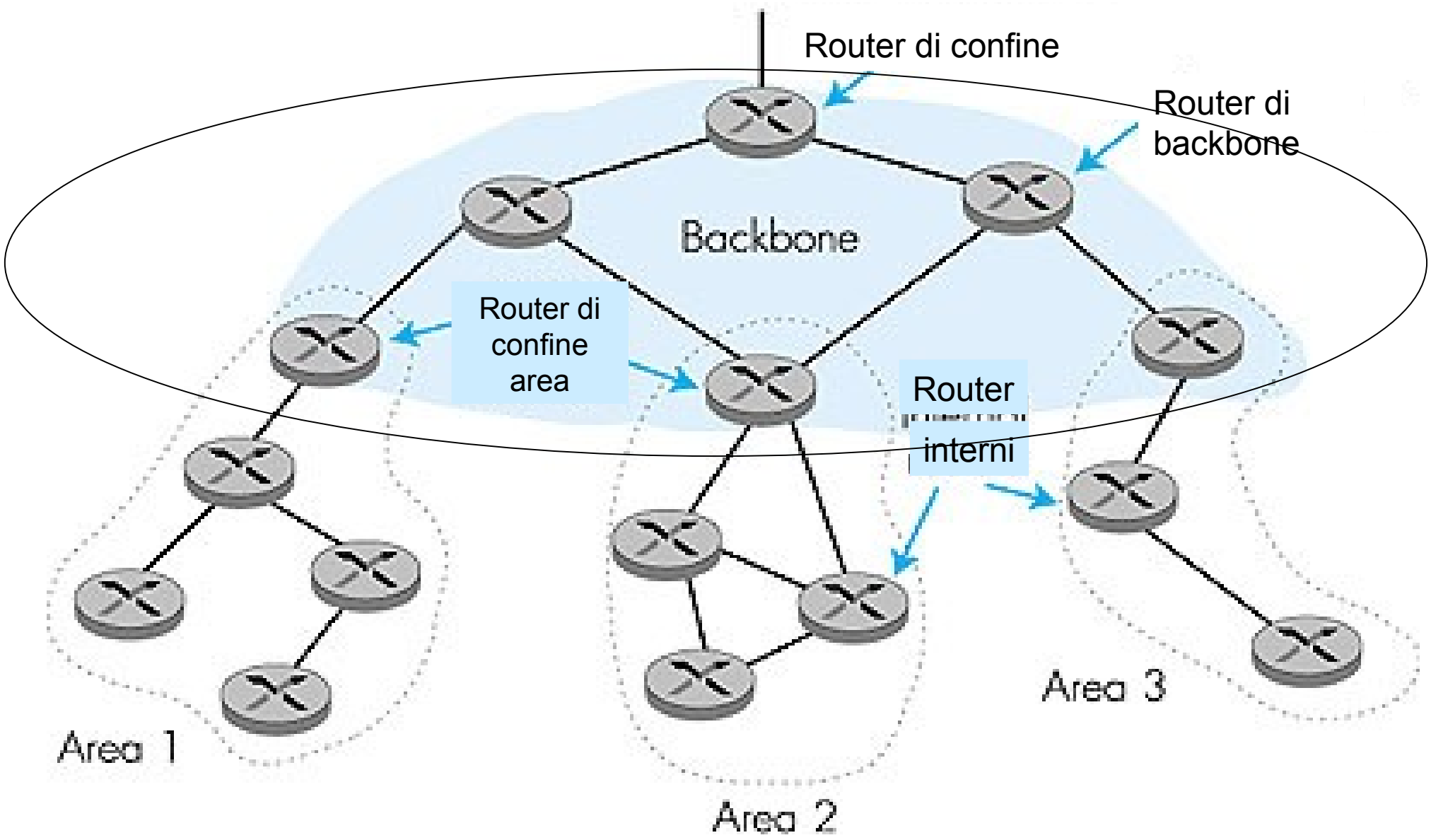
Struttura della gerarchia OSPF

- **Consente una suddivisione di grandi AS in aree così da avere una gerarchia interna a due livelli:**
  - aree locali
  - area backbone
- **Si usa l'algoritmo Link State all'interno di ogni area**

Struttura della gerarchia OSPF

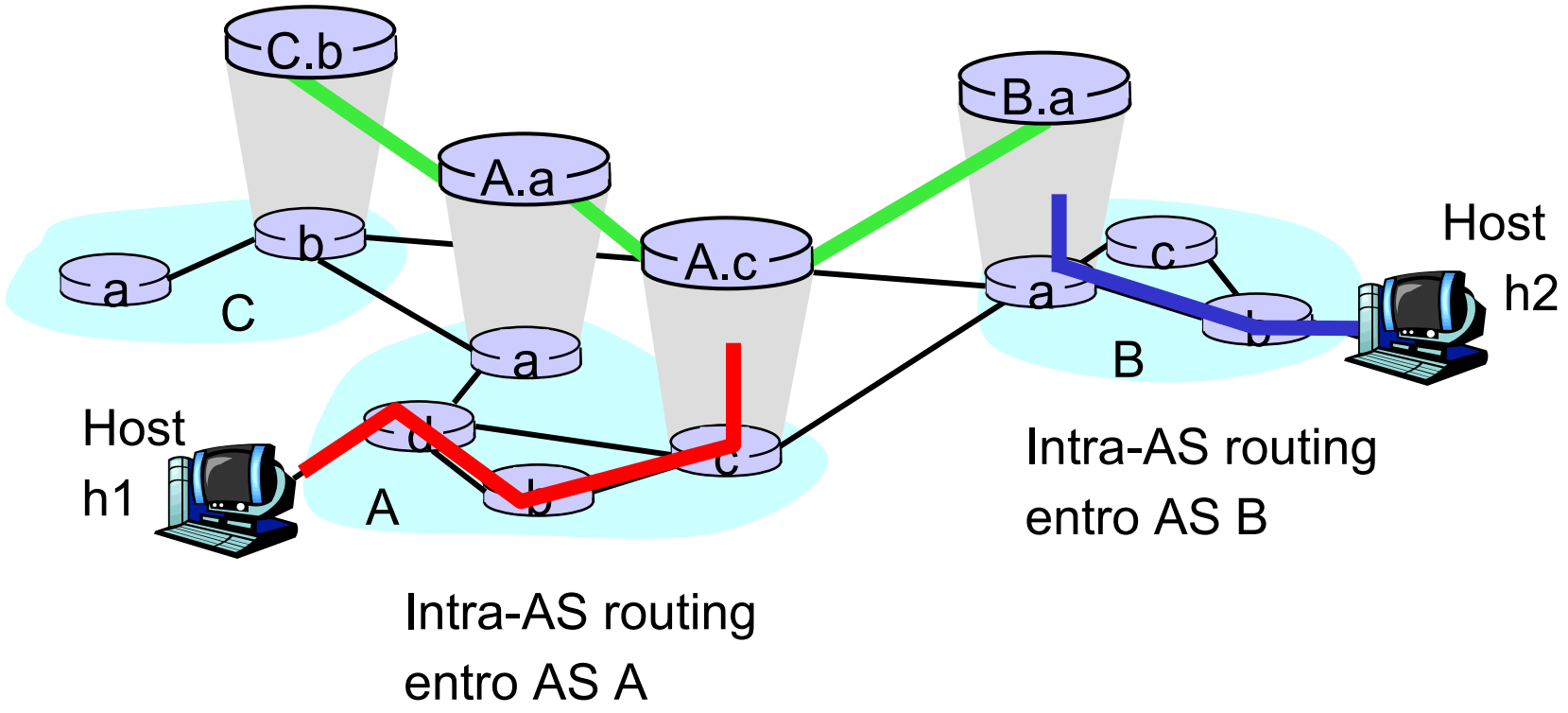
- **Area backbone**
  - Instrada il traffico tra le aree del sistema autonomo
  - Contiene tutti i router di confine
- **Router di confine: gestiscono l'instradamento verso altri AS**
  → implementano l'algoritmo di routing inter-AS
- **Router di backbone (non di confine): effettuano l'instradamento all'interno dell'area backbone**
- **Router di confine area: comunicano i percorsi verso altre aree locali dell'AS ai router di quell'area**

Sistema autonomo OSPF gerarchico

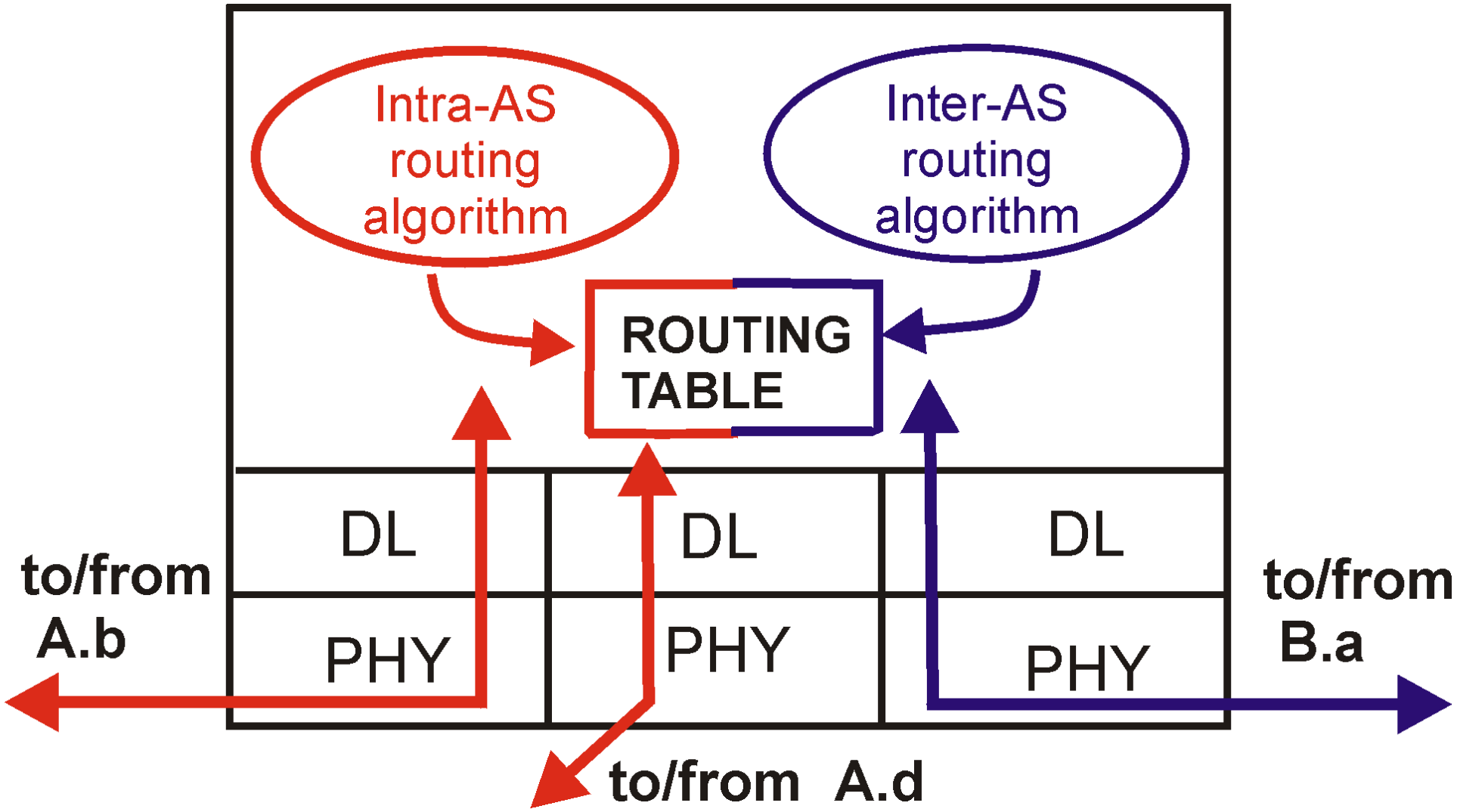


Connettere diversi AS

I router di confine (*gateway router*) hanno la responsabilità di inoltrare pacchetti a destinazioni esterne all'AS



Architettura di un gateway router



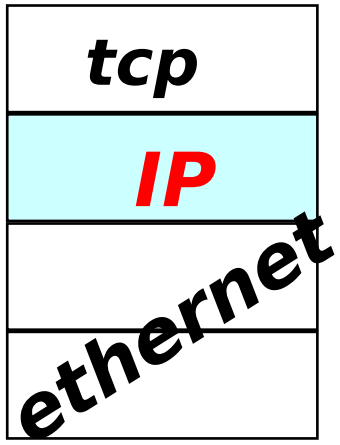
Modulo 12: Protocollo ICMP

protocolli di supporto

protocolli che offrono servizi

portano “dati utente”
le loro pdu vengono imbustate” in accordo alla pila iso-osi

esempi:

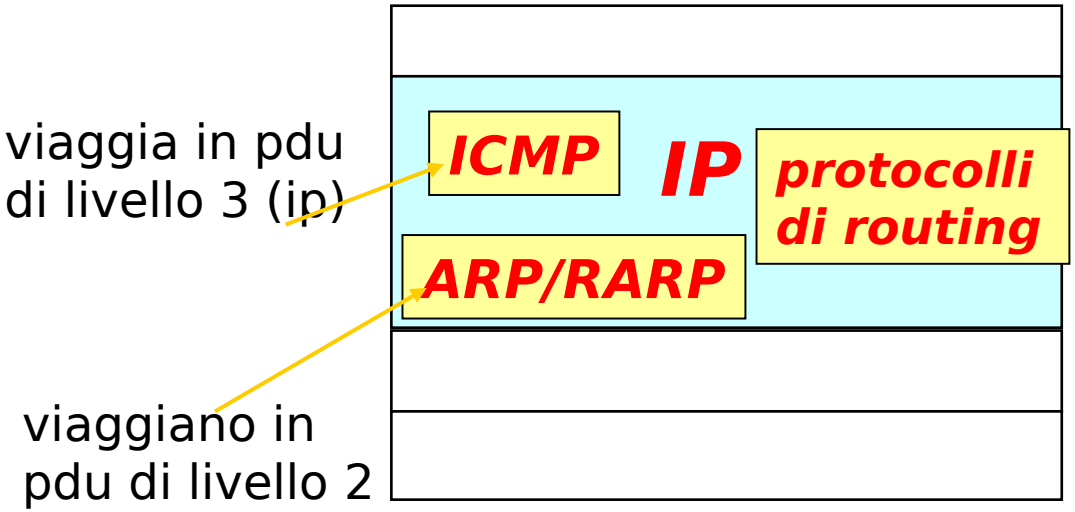


protocolli di supporto

portano dati “di controllo” e non offrono direttamente servizi

*la loro posizione nella pila è **indipendente** da come vengono imbustate le pdu*

esempi:



Internet Control Message Protocol (ICMP), RFC 792

- **Scopi**
  - Notifica situazioni di errore o anomalie
  - Supporta “debugging” interattivo della rete
- **ICMP è funzionale ad IP e viaggia in pacchetti IP**
- **ICMP porta informazioni di controllo e di notifica di errori. Non porta dati. I pacchetti ICMP viaggiano all’interno dei pacchetti IP.**
- **Le funzionalità fornite da ICMP sono accessorie al livello 3, quindi ICMP è considerato un protocollo di livello 3. ICMP è sempre presente a fianco di IP.**

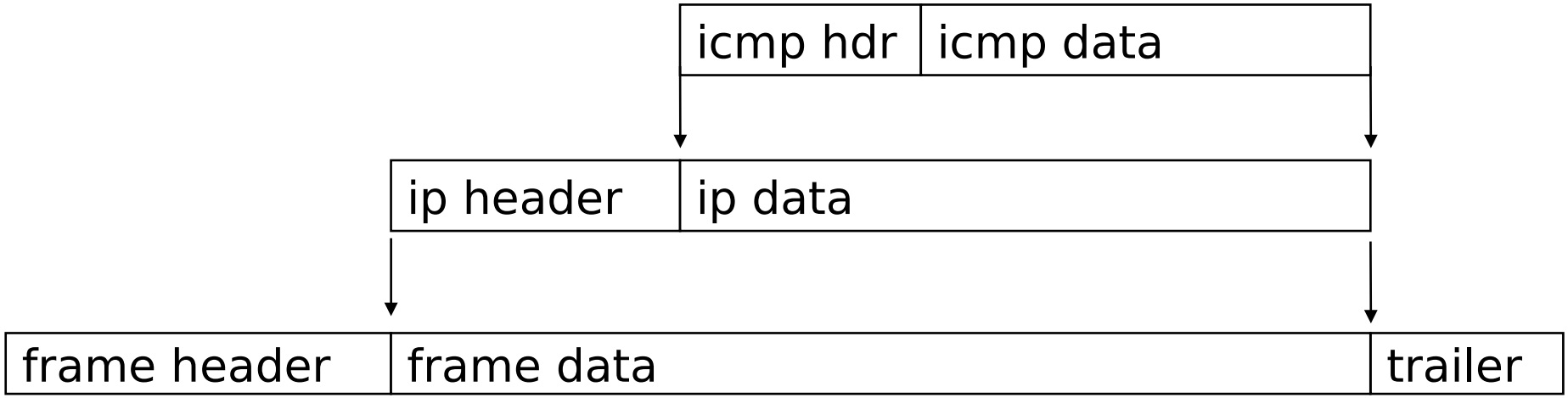
Internet Control Message Protocol (ICMP), RFC 792

- **ICMP interviene quando c'è una anomalia nel processo di instradamento e una condizione di errore deve essere notificata al mittente del pacchetto. Poiché i pacchetti ICMP devono venire instradati, non è detto che tale notifica pervenga al mittente. In pratica tale situazione è abbastanza rara.**
- **ICMP è utile all'amministratore della rete per testare la raggiungibilità del livello IP di un host remoto con un dato indirizzo (comando `ping`). Un uso molto furbo di ICMP ci permetterà di capire che strada fanno (per quali router passano) i nostri pacchetti (comando `traceroute`)**

ICMP
Internet Control Message Protocol

- **ICMP è un protocollo per lo scambio di messaggi di controllo**
- **ICMP usa IP per inviare i propri messaggi**
- **I messaggi ICMP sono tipicamente rivolti al layer IP dell'altro host, non all'utente**

Relazione tra ICMP e IP



regola 1: nessun messaggio ICMP viene generato a seguito ad eventuali errori rilevati su messaggi ICMP
regola 2: se il pacchetto viene frammentato solo il primo frammento può generare messaggi di errore ICMP
regola 3: i broadcast e multicast non generano ICMP

ICMP Message Types

- **Echo Request**
- **Echo Response**
- **Destination Unreachable**
- **Time Exceeded**
- **Redirect (route change)**
- **Molti altri messaggi sono disponibili**

Messaggi di errore

Questi messaggi di errore seguono un pacchetto scartato:

- **TIME_ EXCEEDED (tempo scaduto)**
  - il pacchetto ha TTL=0
- **DESTINATION_UNREACHABLE**
  - un gateway vede la rete destinazione a distanza infinita (net unreachable)
  - l'host non risponde ad una chiamata ARP (host unreachable)
  - l'host destinazione non conosce il protocollo nel pacchetto (protocol unreachable)
  - il pacchetto non può essere frammentato (fragmentation needed and DF set)

Messaggi di errore

Questi messaggi di errore seguono un pacchetto scartato:

- **PARAMETER_PROBLEM (problema con i parametri).**
  - Il gateway non riesce ad interpretare il pacchetto ricevuto a causa di un valore errato, possibile errore software
- **SOURCE_QUENCH obsoleto (rallentamento, soffocamento della sorgente)**
  - congestione di un gateway intermedio
  - host destinazione lento nell'acquisizione

Messaggi di informazione

- **ECHO_REQUEST e REPLY (richiesta di echo e relativa risposta)**
  - controllo di raggiungibilità di un host
- **TIMESTAMP e TIMESTAMP_REPLY come ECHO più informazioni su orario invio**
  - misura di velocità del collegamento
  - sincronizzazione (approssimativa) dell'ora di sistema

Messaggi di informazione

- **REDIRECT (ridireziona). Un router intermedio si accorge che il prossimo router cui dovrebbe inoltrare il pacchetto sta sulla stessa LAN del mittente**
- **Comandi ping e traceroute usano messaggi ICMP**

Routing redirect

- **Si supponga che HostA debba inviare a HostB un pacchetto. Poiché i due host non sono nella stessa subnet, HostA invia il pacchetto al suo default gateway (es., R1).**
- **R1 ha nella sua tabella di instradamento una regola che indica di instradare i pacchetti destinati alla rete 132.16.0.0/16 tramite R2. Poiché R2 è raggiungibile anche da HostA, R1 comunica questo fatto a HostA tramite un messaggio di redirect.**
- **HostA invia i successivi pacchetti destinati a HostB tramite R2 tramite un meccanismo di caching.**
- **Nota: contrariamente a quanto indicato dalla RFC 1122 (standard), non tutti gli end-system considerano le indicazioni di redirect.**
- **Ulteriori approfondimenti in: RFC 792, 1812, 1122**

ping

```
<patrigna@pascal ~> ping wilma.cs.brown.edu
 PING wilma.cs.brown.edu: (128.148.19.15): 56 data bytes
64 bytes from 128.148.19.15: icmp_seq=1 ttl=239 time=1736 ms
64 bytes from 128.148.19.15: icmp_seq=2 ttl=239 time=1507 ms
64 bytes from 128.148.19.15: icmp_seq=6 ttl=239 time=1209 ms
64 bytes from 128.148.19.15: icmp_seq=8 ttl=239 time=762 ms
64 bytes from 128.148.19.15: icmp_seq=9 ttl=239 time=1235 ms
64 bytes from 128.148.19.15: icmp_seq=11 ttl=239 time=1566 ms
64 bytes from 128.148.19.15: icmp_seq=13 ttl=239 time=586 ms
64 bytes from 128.148.19.15: icmp_seq=14 ttl=239 time=352 ms
64 bytes from 128.148.19.15: icmp_seq=22 ttl=239 time=910 ms
^C
----wilma.cs.brown.edu PING Statistics----
29 packets transmitted, 9 packets received, 68% packet loss
round-trip min/avg/max = 352/1095/1736 ms
<patrigna@pascal ~>
```

ping

- **Invia una successione di pacchetti ICMP ECHO_REQUEST e attende la relativa risposta ECHO_REPLY**
- **Misura il tempo che intercorre tra l'invio e la ricezione di ogni pacchetto e riporta semplici statistiche**
- **Il comando ping fa parte della dotazione standard di tutte le macchine sia Unix che Windows anche se con sintassi leggermente diverse.**

ping

- **L'indirizzo wilma.cs.brown.edu viene risolto in un indirizzo IP normale che dall'output è 128.148.19.15**
- **In questo output alcuni pacchetti risultano persi, esattamente il 68%**
- **Ulteriori informazioni: manuale in linea di ping (sui sistemi Unix: man ping)**

ping su indirizzo broadcast

```
<patrigna@pascal ~>ping 193.204.162.255
PING 193.204.162.255: (193.204.162.255): 56 data bytes
64 bytes from 193.204.162.32: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 193.204.162.7: icmp_seq=0 ttl=255 time=1 ms (DUP!)
64 bytes from 193.204.162.9: icmp_seq=0 ttl=255 time=5 ms (DUP!)
64 bytes from 193.204.162.113: icmp_seq=0 ttl=255 time=6 ms (DUP!)
64 bytes from 193.204.162.114: icmp_seq=0 ttl=255 time=7 ms (DUP!)
64 bytes from 193.204.162.20: icmp_seq=0 ttl=255 time=8 ms (DUP!)
64 bytes from 193.204.162.196: icmp_seq=0 ttl=60 time=9 ms (DUP!)
64 bytes from 193.204.162.152: icmp_seq=0 ttl=255 time=10 ms (DUP!)
64 bytes from 193.204.162.215: icmp_seq=0 ttl=64 time=10 ms (DUP!)
64 bytes from 193.204.162.189: icmp_seq=0 ttl=64 time=11 ms (DUP!)
64 bytes from 193.204.162.131: icmp_seq=0 ttl=255 time=11 ms (DUP!)
64 bytes from 193.204.162.128: icmp_seq=0 ttl=64 time=11 ms (DUP!)
64 bytes from 193.204.162.22: icmp_seq=0 ttl=60 time=12 ms (DUP!)
64 bytes from 193.204.162.108: icmp_seq=0 ttl=64 time=12 ms (DUP!)
64 bytes from 193.204.162.194: icmp_seq=0 ttl=63 time=13 ms (DUP!)
....
```

Tutti rispondono al ping!

Attenzione il trattamento degli indirizzi broadcast non è uniforme nei vari sistemi. I sistemi Windows non permettono di lanciare un ping di questo tipo.

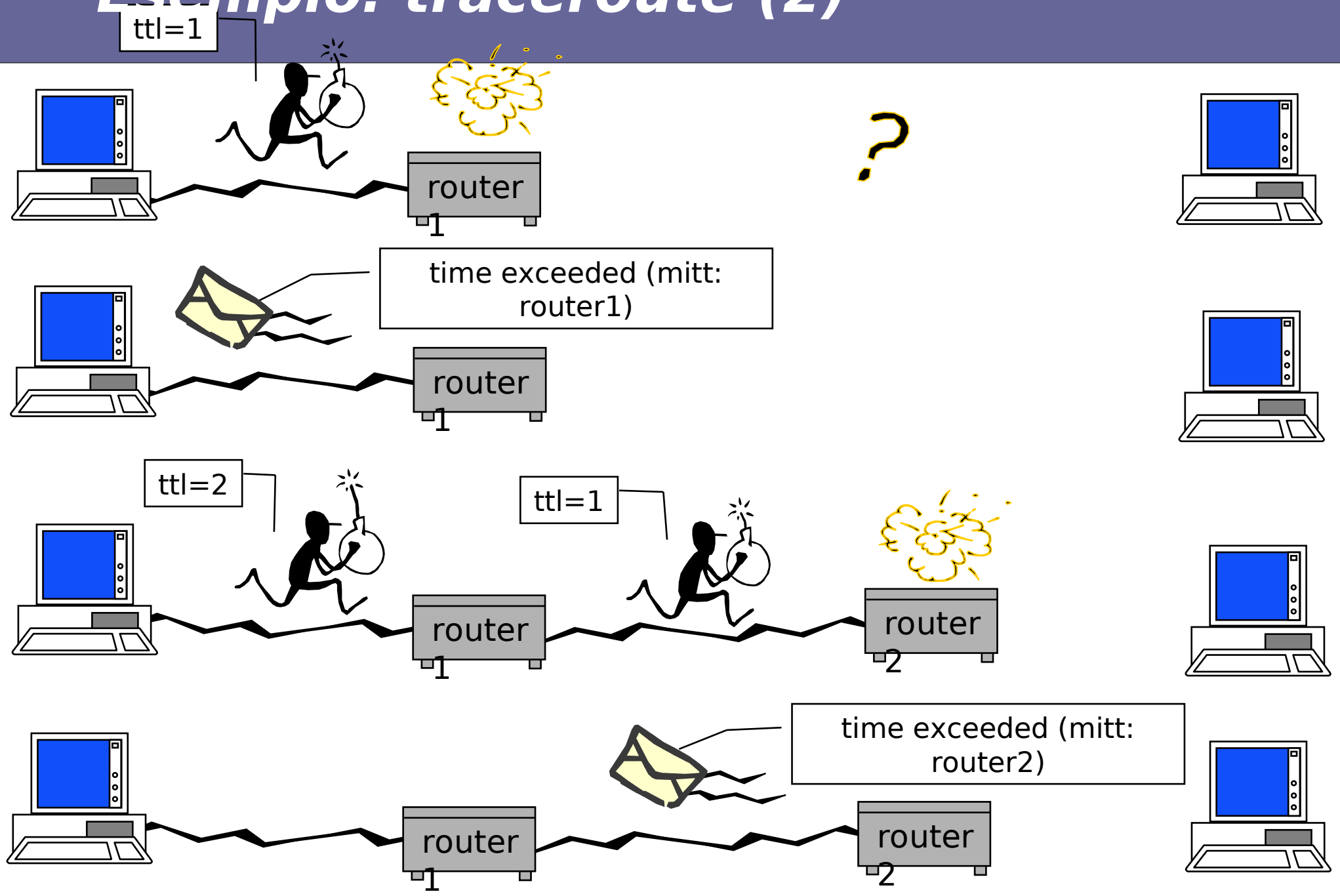
traceroute

```
<utente@pascal ~>traceroute wilma.cs.brown.edu
traceroute to wilma.cs.brown.edu (128.148.19.15), 30 hops max, 40 byte packets
1  gw1.fis.uniroma3.it (193.204.160.1)  3 ms  3 ms  2 ms
2  141.108.132.1 (141.108.132.1)  832 ms  967 ms  402 ms
3  mp4rm1.roma1.infn.it (141.108.127.6)  267 ms  106 ms  417 ms
4  atm-garrten-rm.infn.it (192.135.31.5)  100 ms  939 ms  839 ms
5  cnafint-ten34.infn.it (192.135.34.21)  1100 ms *  1056 ms
6  mix-serial3-4.Washington.mci.net (204.189.152.161)  618 ms * *
7  * core1-fddi-0.Washington.mci.net (204.70.2.1)  1249 ms *
8  * * *
9  wtn-bbn-nap.Washington.mci.net (206.157.77.218)  766 ms * *
10  * * chicago1-br1.bbnplanet.net (4.0.1.5)  857 ms
11  * * *
12  * boston1-br1.bbnplanet.net (4.0.2.245)  846 ms *
13  boston1-br2.bbnplanet.net (4.0.2.250)  680 ms * *
14  * * boston1-mr4.bbnplanet.net (4.0.44.19)  648 ms
15  providence-cr1.bbnplanet.net (4.0.45.106)  416 ms providence-cr1.bbnplanet.net (4.0.45.102)  1298 ms *
16  brown.bbnplanet.net (131.192.32.2)  1444 ms  615 ms  802 ms
17  * * *
18  * ftp.cs.brown.edu (128.148.19.15)  834 ms  435 ms
<utente@pascal ~>
```

Esempio: `traceroute`

- **Si ipotizzi di inviare un pacchetto, ad esempio un ECHO_REQUEST con TTL=*n* “basso”. Se il destinatario è troppo lontano l’ultimo router risponderà con un TIME_EXCEEDED, se il destinatario risponde con ECHO_REPLY allora è raggiungibile con *n* hop**
- **In realtà alcuni `traceroute` non utilizzano gli ECHO_REQUEST se non gli viene chiesto esplicitamente ma altri protocolli (UDP). Inoltre manda 3 pacchetti per ogni valore di *n*. Se non si osserva un TIME_EXCEEDED entro un tempo prestabilito `traceroute` visualizza un asterisco “*”**
- **E’ anche possibile che il pacchetto venga duplicato e più di un router risponda con un TIME_EXCEEDED, oppure che risponda un router diverso per ciascuna delle tre prove, nel qual caso `traceroute` visualizza gli indirizzi di tutti i router che hanno risposto**
- **Il manuale in linea di `traceroute` (nella versione GNU in Linux RedHat 6.0) è estremamente interessante e mostra anche esempi di strani comportamenti da parte dei router**

Esempio: traceroute (2)



- **Il comando `traceroute` permette di capire per quali router passano i pacchetti IP quando sono diretti ad una data destinazione.**
- **`traceroute` è compreso nella dotazione standard di tutte i sistemi Unix.**
- **Nei sistemi Windows il comando ha il nome `tracert` e sintassi leggermente differente.**
- **`traceroute` trova utilizzo anche nell'ambito dell'amministrazione della rete per isolare guasti o incoerenze nelle tabelle di routing.**
- **Dal punto di vista didattico la sua utilità è di mostrare il percorso dei pacchetti svelando un po' della topologia nascosta di Internet.**