

16 Febbraio 2024

La Galassia del Routing IP

Il cuore dell'Internet



I[^] puntata - Le basi

Tiziano Tofoni

Note di *Copyright*

- Questo insieme di diapositive è protetto dalle leggi sul *copyright* e dalle disposizioni dei trattati internazionali. Il titolo ed i *copyright* relativi alle diapositive (ivi inclusi, ma non limitatamente, ogni immagine, fotografia, animazione, video, audio, musica e testo), in accordo con gli artt. 12 e seguenti della Legge 633/1941, **sono di proprietà dell'autore Tiziano Tofoni** (di seguito 'l'autore').
- Le diapositive **possono essere utilizzate esclusivamente per scopi di studio nell'ambito dei corsi tenuti dall'autore**.
- Ogni altra utilizzazione o riproduzione (ivi incluse, ma non limitatamente, le riproduzioni su supporti ottici/magnetici, su reti di calcolatori o stampate) in toto o in parte **è vietata, se non esplicitamente autorizzata per iscritto, a priori, da parte dell'autore**.
- L'informazione contenuta in queste diapositive è ritenuta essere accurata alla data della pubblicazione. Essa è fornita per scopi meramente didattici e non per essere utilizzata in progetti di impianti, prodotti, reti, ecc. In ogni caso essa è soggetta a cambiamenti senza preavviso. **L'autore non si assume alcuna responsabilità per il contenuto di queste diapositive** (ivi incluse, ma non limitatamente, la correttezza, completezza, applicabilità, aggiornamento dell'informazione).
- In ogni caso non può essere dichiarata conformità all'informazione contenuta in queste diapositive.
- In ogni caso **questa nota di *copyright* non deve mai essere rimossa e deve essere riportata anche in utilizzi parziali**.

Di cosa parlerò ...

#1

A cosa serve il routing IP

#2

Concetti fondamentali

#3

Routing statico e dinamico

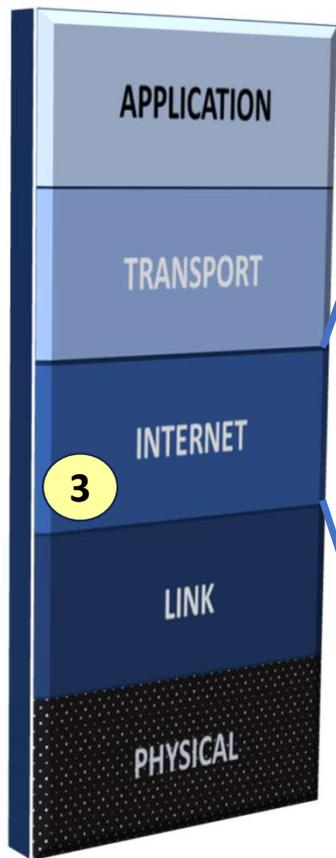
#4

Protocolli di routing *Distance/Path Vector* e *Link State*

#5

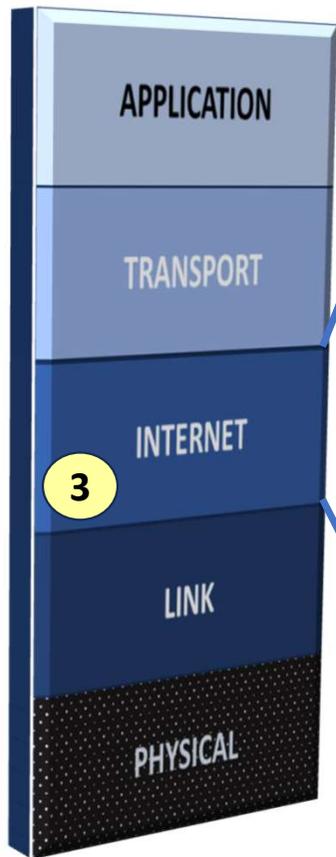
Routing *multicast*

Prologo: il Livello 3 della Internet Protocol Suite



- Il Livello 3 (*network* nel linguaggio OSI, *Internet* nel linguaggio TCP/IP) si occupa di far arrivare i pacchetti a destinazione, attraversando una o più Reti, facendo salti (*hop*) da un nodo (*router*) all'altro
- Per farlo deve determinare i percorsi appropriati **evitando di sovraccaricare in modo disequilibrato diverse parti della Rete**
- Il suo lavoro è più complesso quando la sorgente e la destinazione sono in reti diverse (***Autonomous Systems, AS*** - sistemi autonomi)
- La determinazione dei percorsi può essere definita **manualmente** oppure demandata a particolari **algoritmi dinamici**

Prologo: il Livello 3 della *Internet Protocol Suite*



Il Livello 3 offre al livello superiore **DUE servizi fondamentali**:

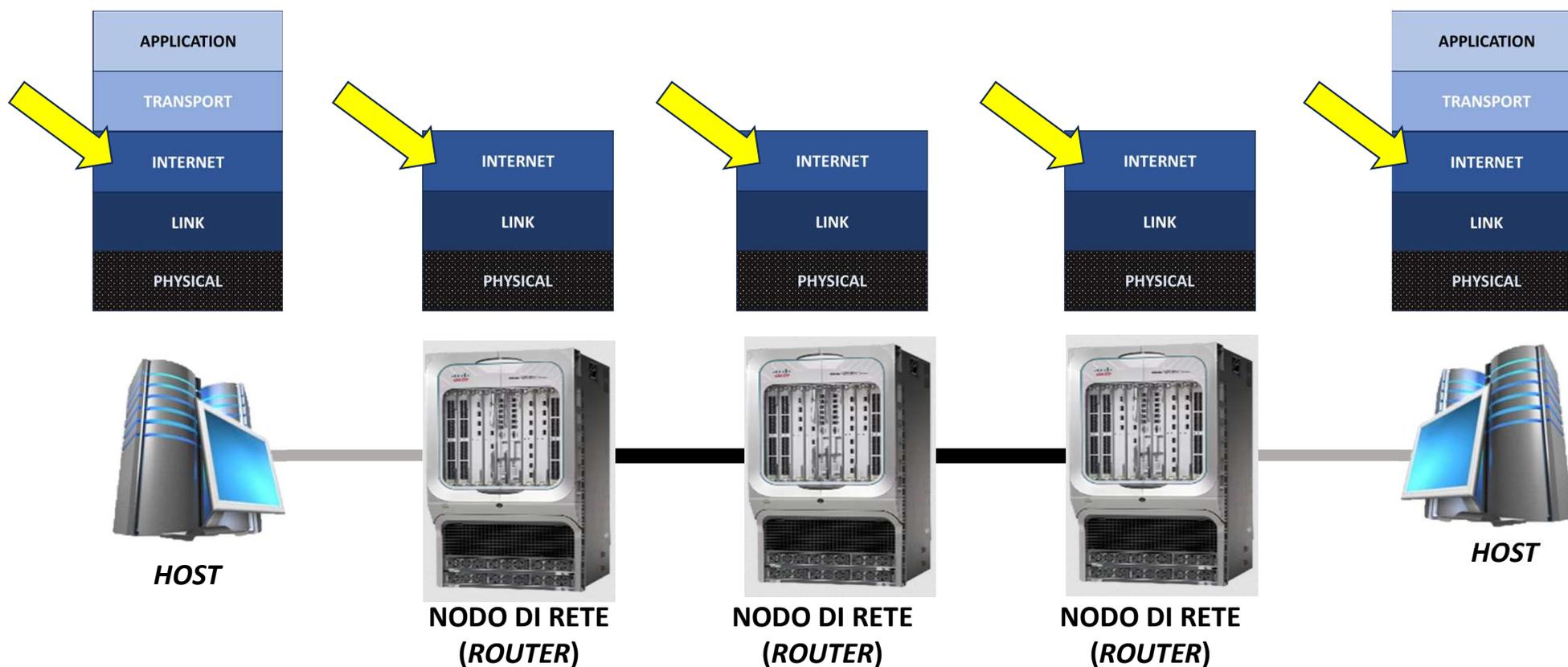
1. indirizzamento (**addressing**): indirizzi logici unici, indipendenti dagli indirizzi della sottorete trasmissiva
2. inoltrò (**forwarding**): capacità di individuare ad ogni nodo la giusta interfaccia di uscita fino ad arrivare all'*host* di destinazione

...per farlo, implementa funzioni di routing: determinare la strada che deve essere percorsa dai segmenti dati che gli vengono affidati dal livello superiore (trasporto)

...e si avvale dei servizi offerti dal sottostante Livello 2

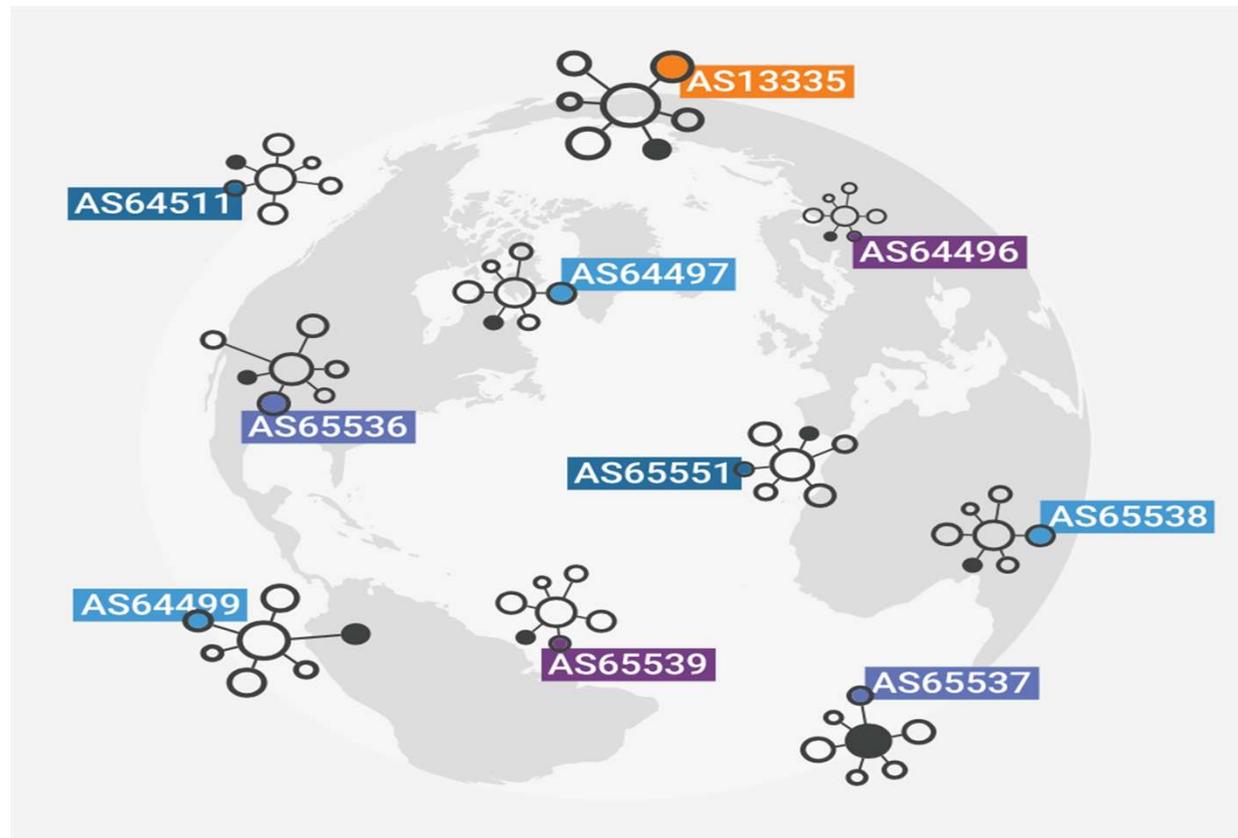
Dove sono implementate le funzioni del Livello 3?

- In ciascun *host* (*client*, *server*, ecc.) e nei **NODI** di rete (i **ROUTER**)



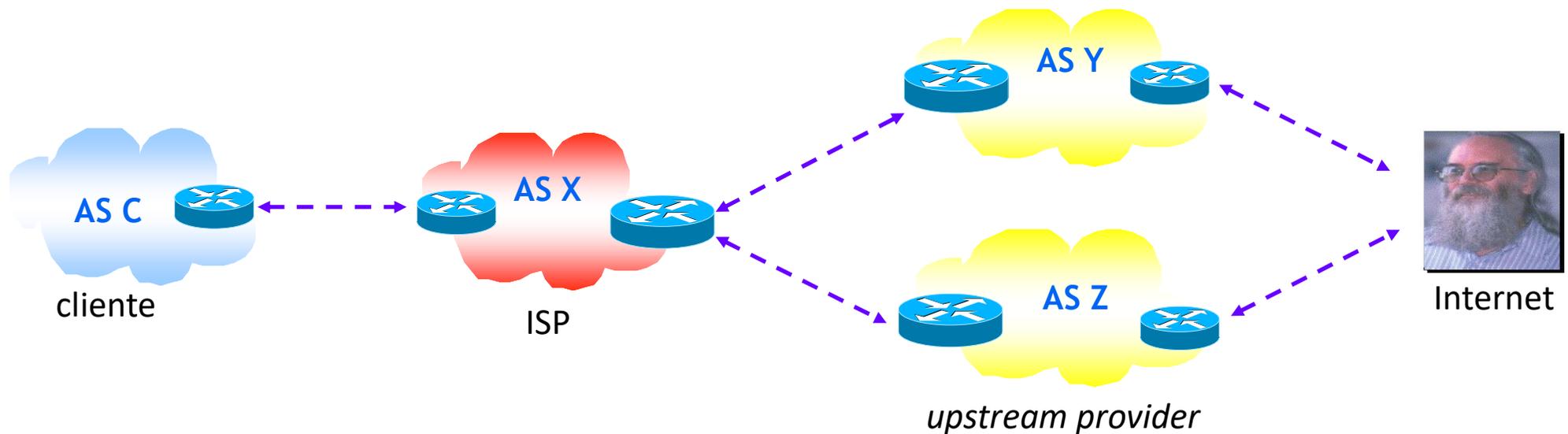
L'internet...

- L'Internet è un insieme di reti (*Autonomous System*) interconnesse tra loro



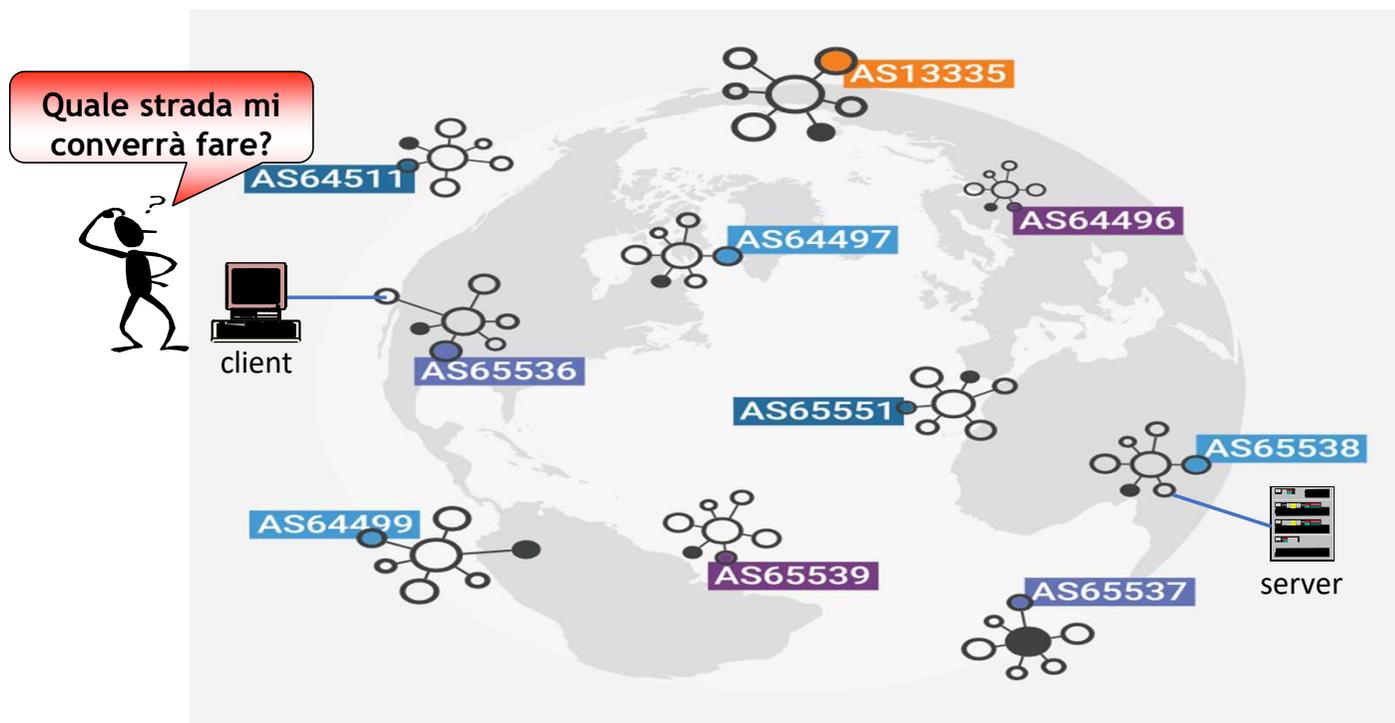
Il concetto di *Autonomous System* (AS)

- Un AS è un insieme di reti amministrato da un singolo ente
- Ogni AS è identificato universalmente da un numero di 16 o 32 bit
 - Esempi: la rete IP di **TIM** ha **AS=3269**, la rete IP di **Fastweb** ha **AS=12874**, la rete **Seabone** di **TI Sparkle** ha **AS=6762**, la rete GIN di **NTT** ha **AS=2914**, ecc.



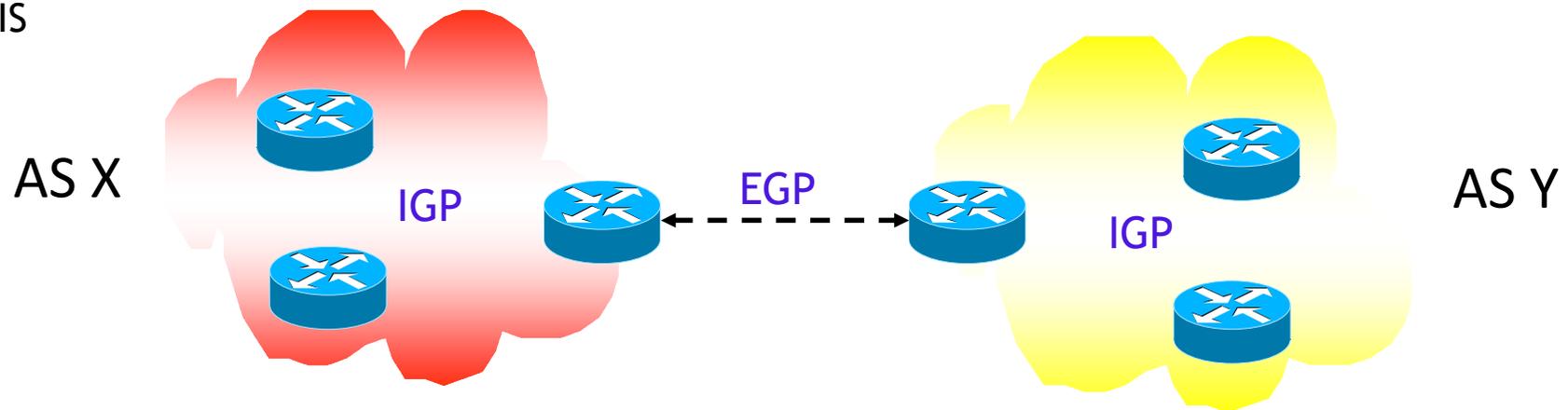
A cosa serve il routing?

Il Routing (*instradamento*) è, in termini generali, il lavoro, tipico del Livello 3, necessario per far arrivare a destinazione i pacchetti IPv4/v6 (le PDU del Livello 3), attraversando i nodi delle reti interconnesse



Routing IGP e Routing EGP

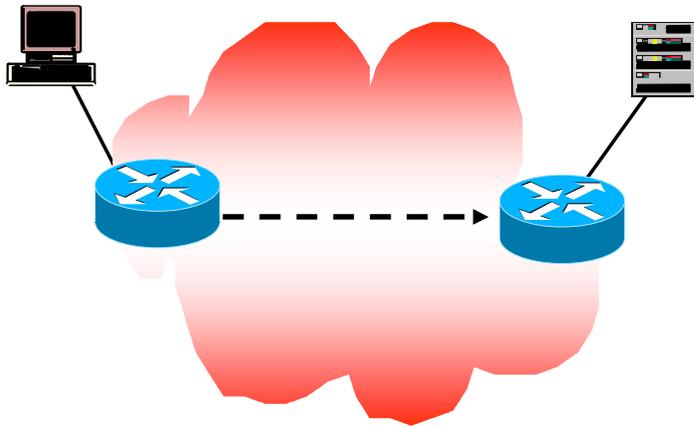
- Gli AS si scambiano informazioni di routing attraverso protocolli di tipo EGP (*)
 - Il BGP è lo standard “de facto” universalmente adottato come **protocollo EGP**
- Al loro interno gli AS scambiano informazioni di routing attraverso protocolli di tipo IGP (*)
 - Esempi: OSPF, IS-IS



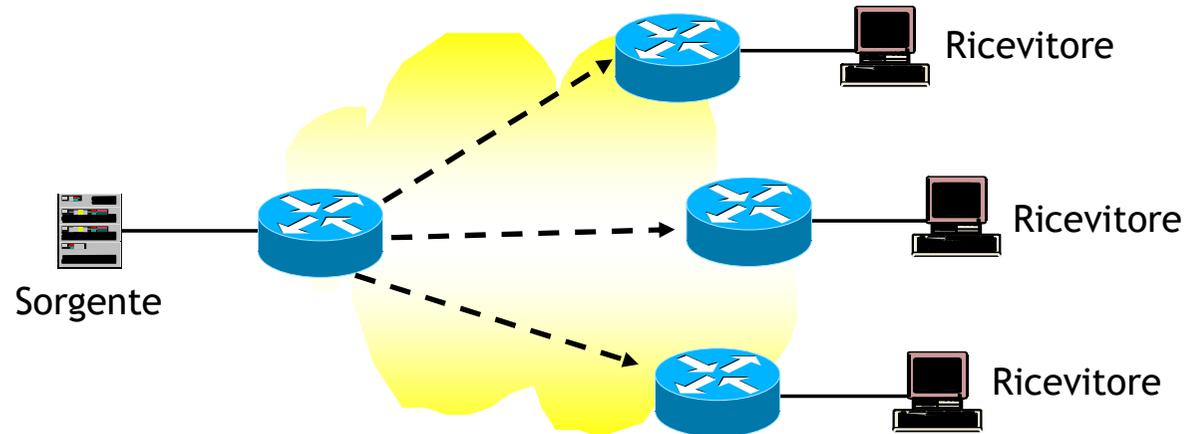
(*) EGP = Exterior Gateway Protocol ; IGP = Interior Gateway Protocol

Routing *unicast* e Routing *multicast*

- Routing *unicast*: regola il traffico da una sorgente a un (unico) destinatario
- Routing *multicast*: regola il traffico da una sorgente a un gruppo di destinatari



Routing **unicast**



Routing **multicast**

I tre paradigmi fondamentali del Routing IP

- *Destination Based Forwarding*

- Il *forwarding* dei pacchetti IP avviene sulla base del solo indirizzo destinazione
- È possibile variare questa tramite opportune configurazioni (es. invia il traffico HTTPS verso un certo *Next-Hop*)

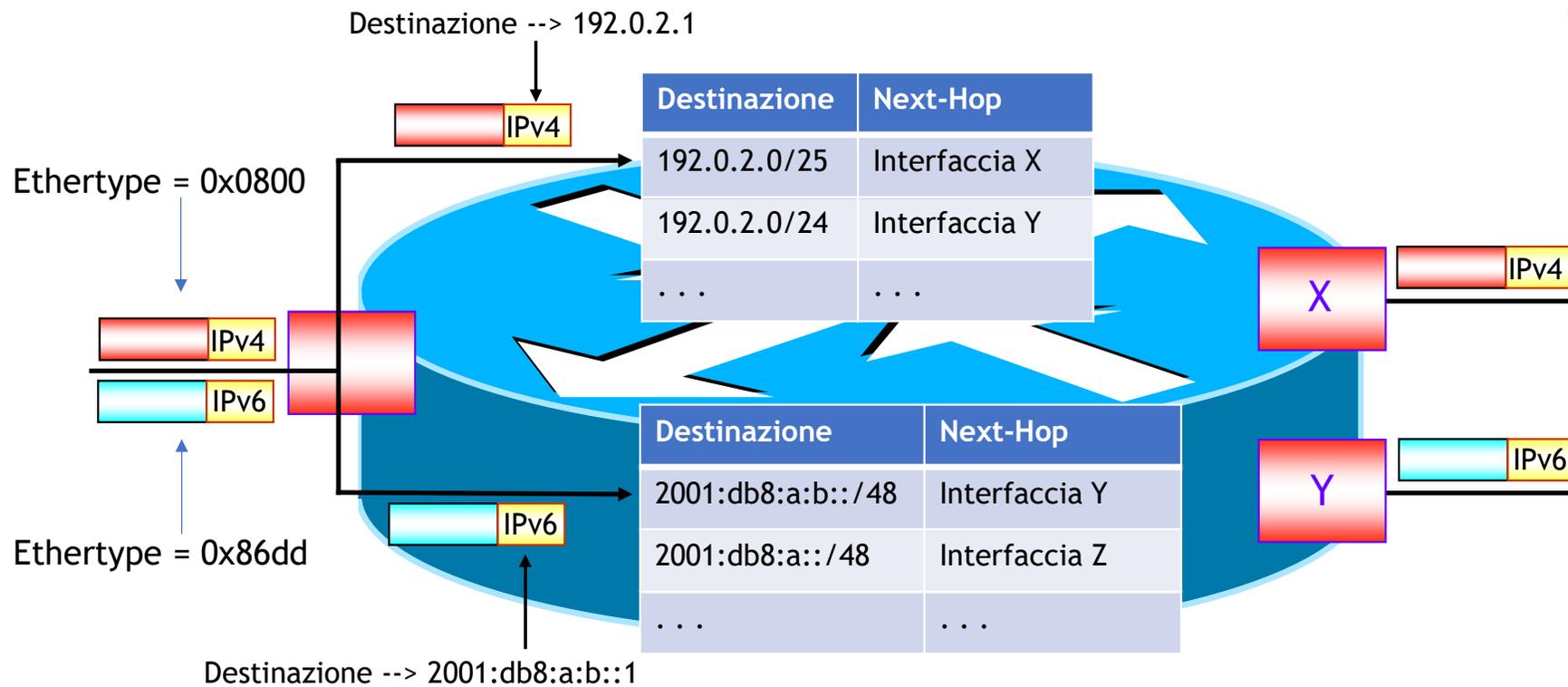
- *Longest Match Prefix*

- Se un pacchetto ha un indirizzo IP destinazione che appartiene a due o più entry della FIB, il *forwarding* dei pacchetti IP avviene utilizzando l'entry più specifico (ossia, quello lunghezza della maschera maggiore)

- *Hop-by-Hop*

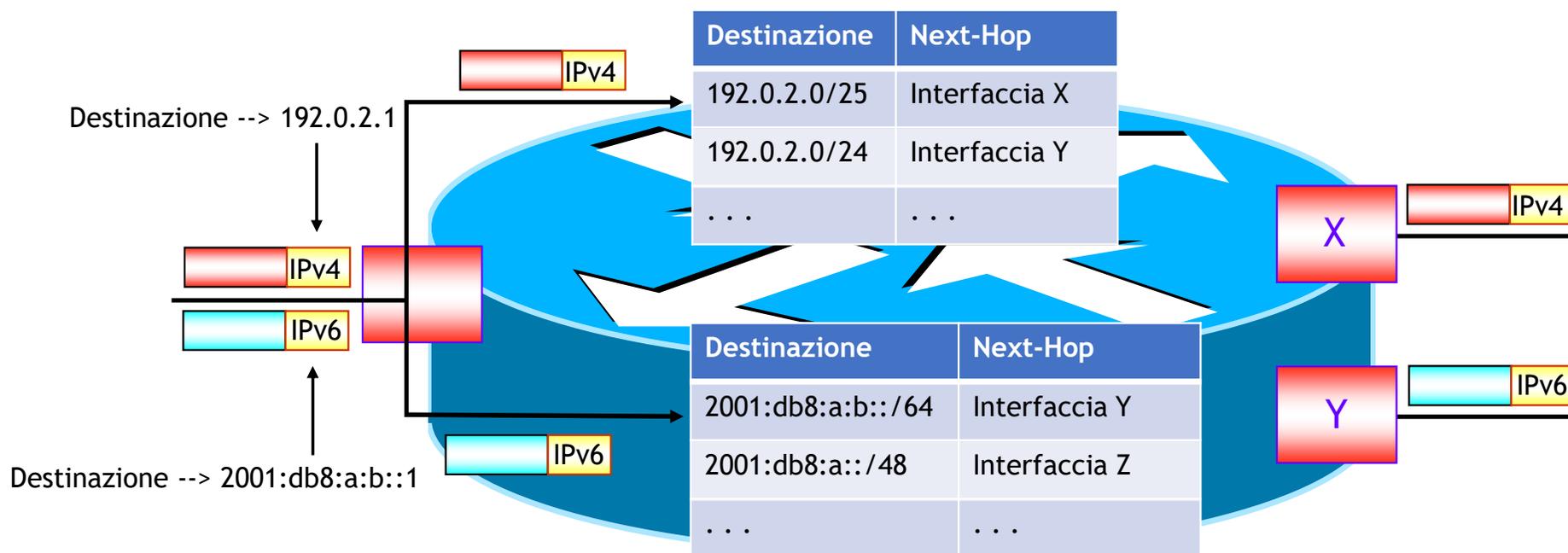
- Il *forwarding* avviene nodo per nodo
- Ciascun nodo attraversato dal pacchetto prende le sue decisioni di routing sulla base del contenuto della FIB

Destination Based Forwarding



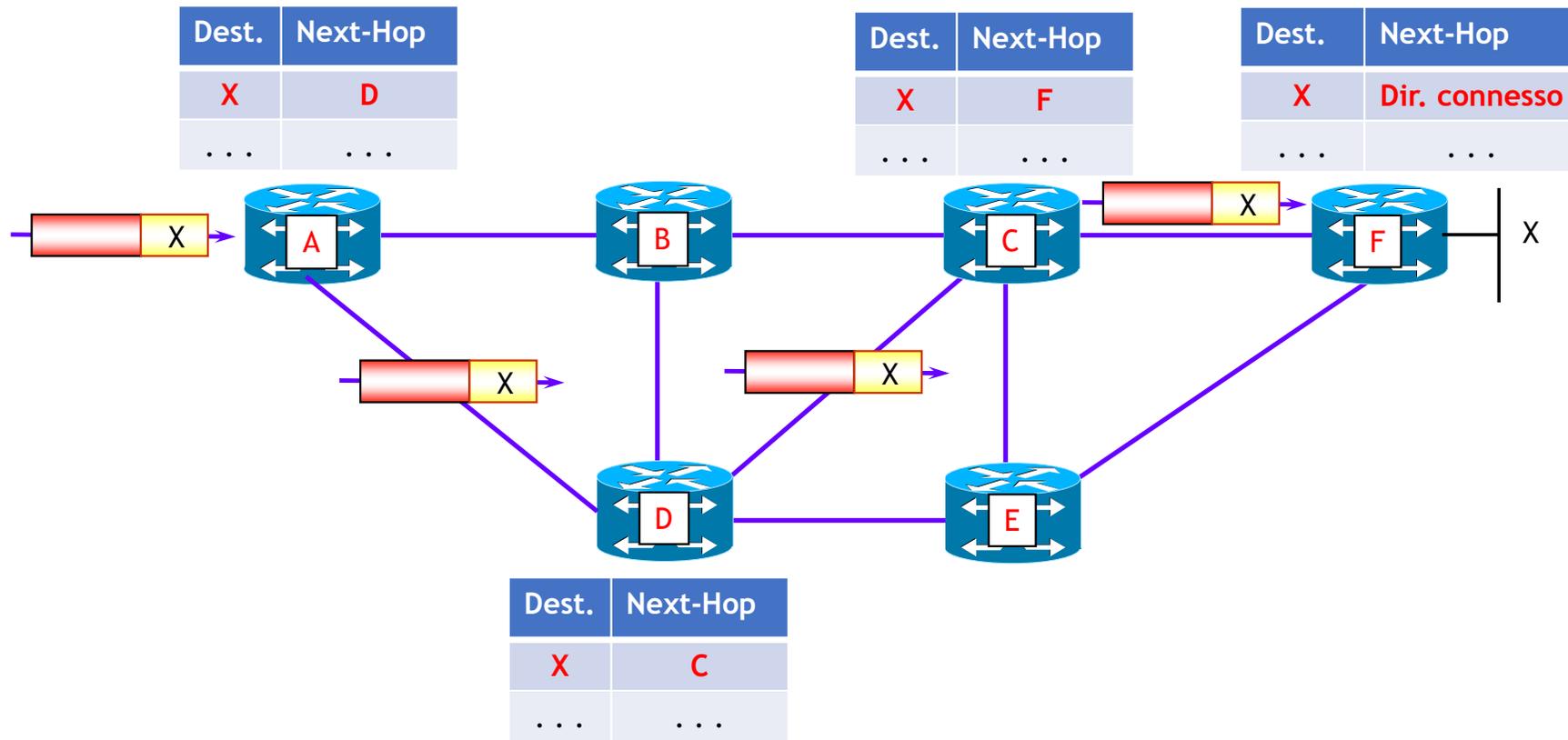
Il *forwarding* dei pacchetti IP avviene sulla base del solo indirizzo destinazione

Longest Match Prefix



Si utilizza l'*entry* più specifico

Hop-by-Hop



Il forwarding avviene **nodo per nodo**

Di cosa parlerò ...

#1

A cosa serve il routing IP

#2

Concetti fondamentali

#3

Routing statico e dinamico

#4

Protocolli di routing *Distance/Path Vector* e *Link State*

#5

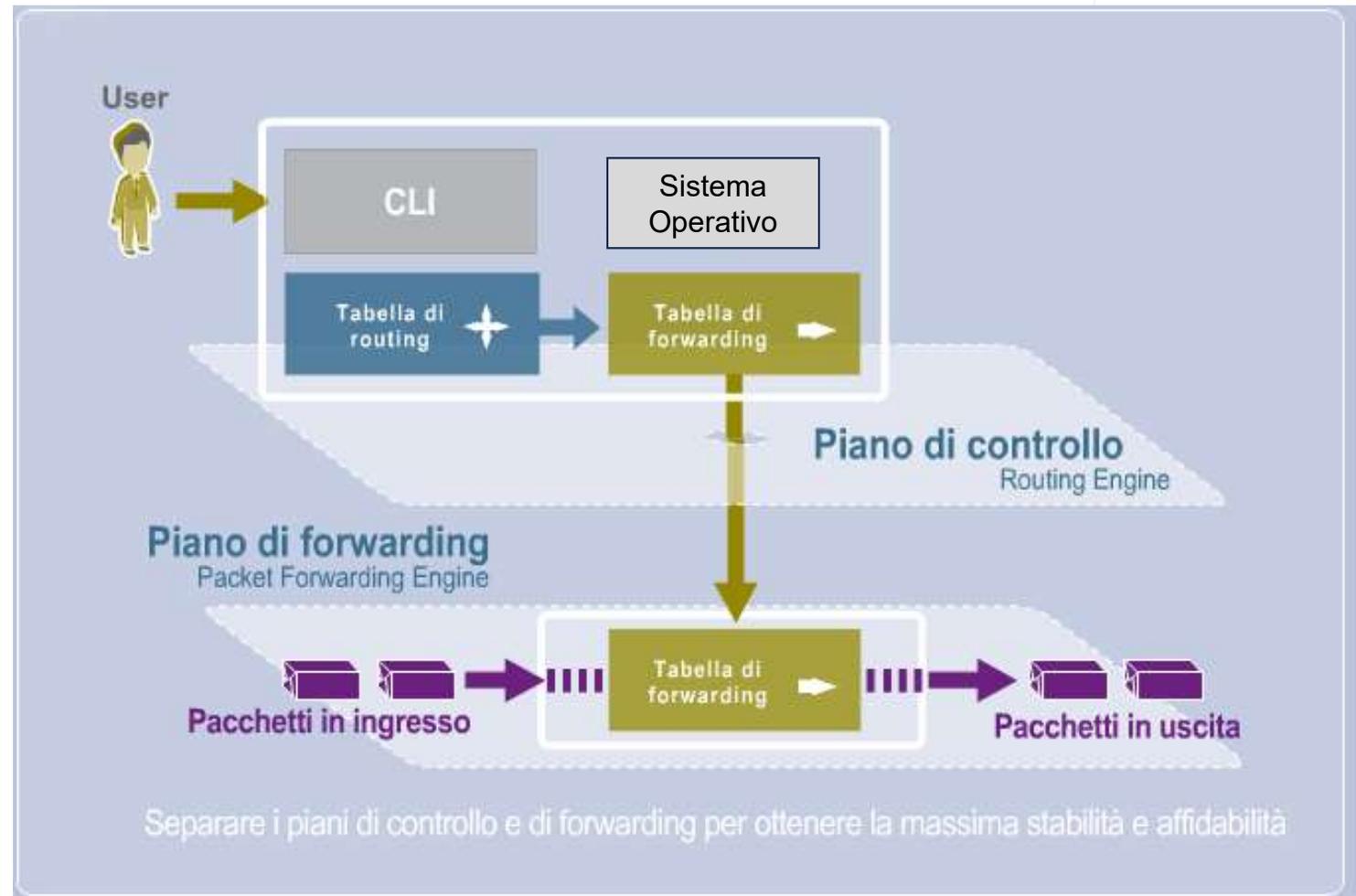
Routing *multicast*

Routing e Forwarding

- Le operazioni fondamentali svolte dal router sono:
 - **Routing**: “impara” la topologia della rete ed inoltra correttamente i pacchetti
 - **Forwarding**: invia i pacchetti ricevuti su una interfaccia d’ingresso verso una interfaccia d’uscita
- I tre elementi fondamentali per l’inoltro di un pacchetto
 1. Quale suite di protocolli si sta utilizzando (TCP/IP, ISO CLNP, ecc.)
 2. L’indirizzo di destinazione
 3. Quale interfaccia di uscita si deve utilizzare (*Next-Hop*)

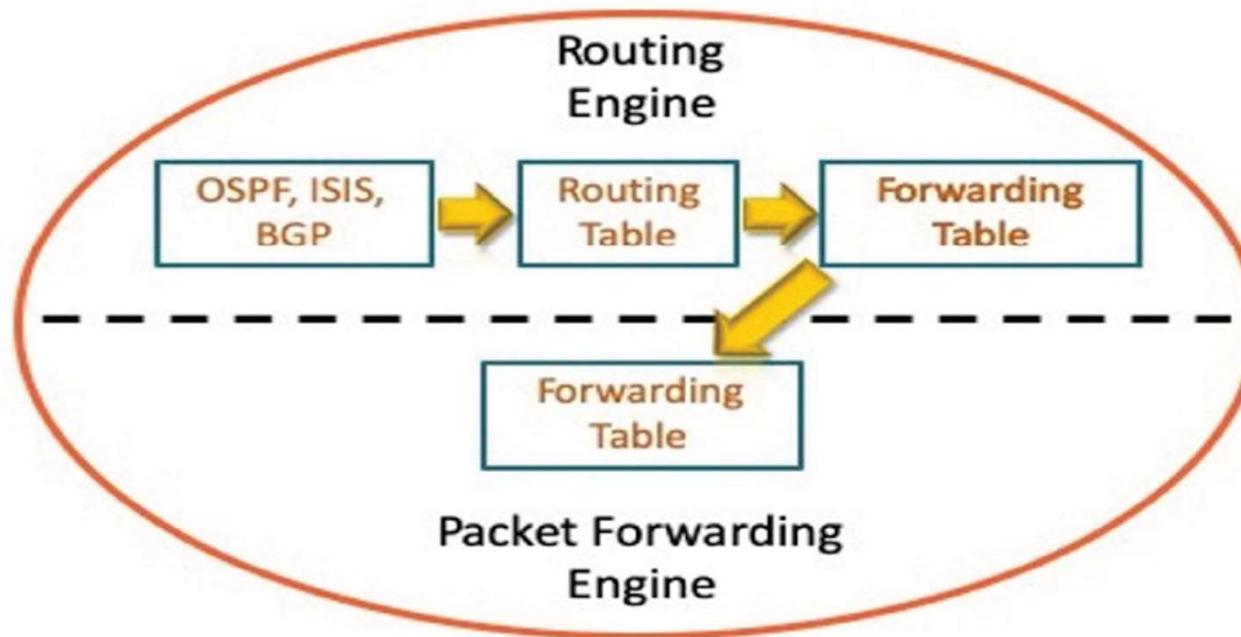
Piano di controllo e piano dati

- Tutti i router di ultima generazione condividono lo stesso criterio generale di progetto: mantenere una **netta separazione fra Piano di Controllo e Piano di Forwarding**



RIB e FIB ...

- La **Tabella di Routing (RIB)** contiene tutti i percorsi appresi tramite i protocolli di routing attivi sulla macchina oppure percorsi definiti staticamente
- La **Tabella di Forwarding (FIB)** contiene i percorsi “migliori” (sulla base di metriche e preferenze) da usare per l’inoltro dei pacchetti.



I campi della Tabella di Routing (RIB)

Sorgente	Rete Destinazione	Costo del percorso	Next-Hop	Interfaccia di uscita
S	0.0.0.0/0	1	Via 10.1.1.1	Gigabitethernet0/0
O	203.0.113.0/24	10	Via 10.1.1.1	Gigabitethernet0/0
B	192.0.2.0/24	0	Via 10.2.2.1	Gigabitethernet0/1
C	198.51.100.0/24	Rete Direttamente Connessa		Gigabitethernet0/2
L	198.51.100.1/32	Indirizzo di una interfaccia locale		Gigabitethernet0/2

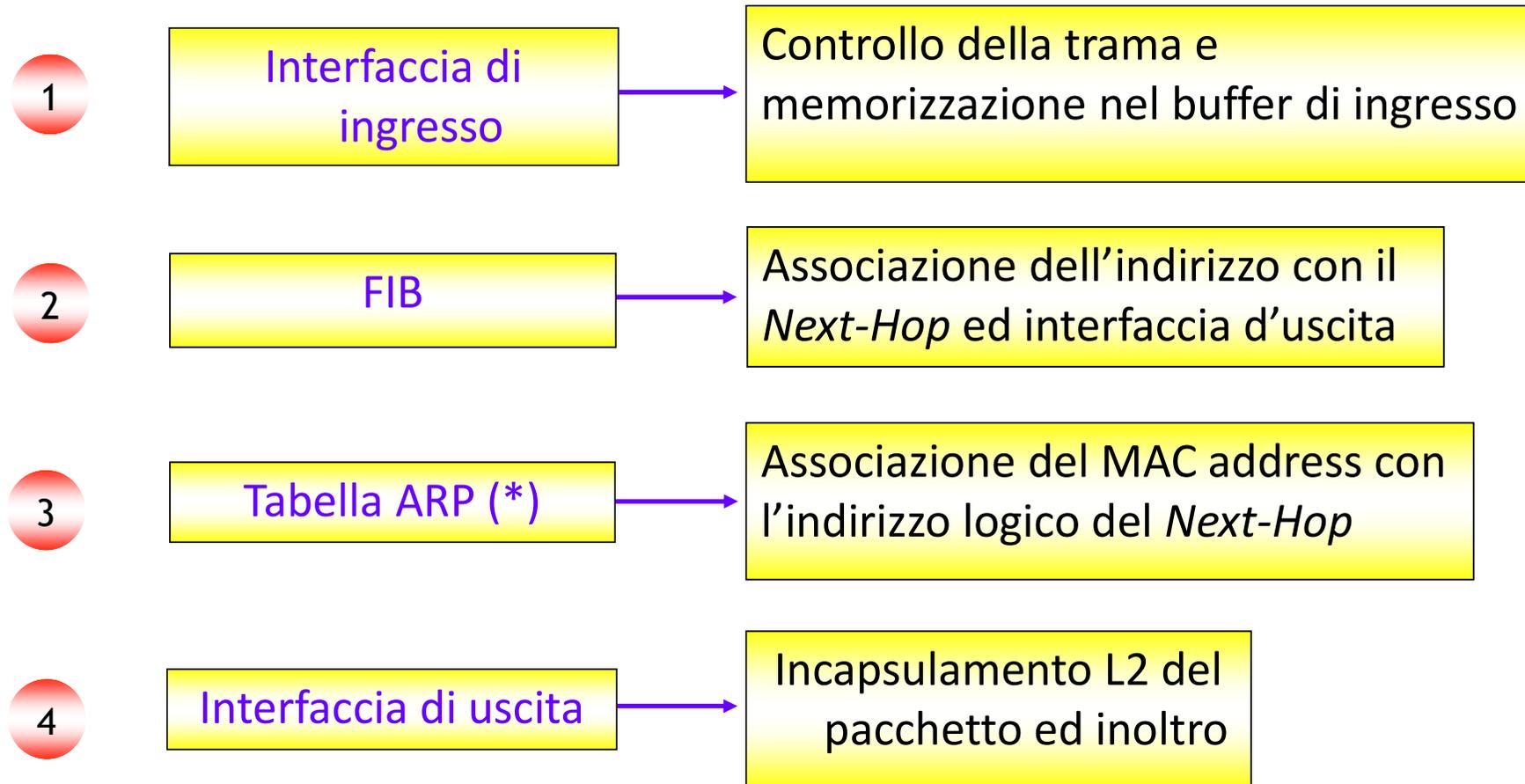
S= Statica – O=OSPF – B=BGP – C=Connessa – L=Locale

La Tabella di Routing (RIB): esempio

```
tt@router> show route table inet.0

inet.0: 49 destinations, 49 routes (49 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.8.0/24      *[Direct/0] 00:11:43 ← Da quanto tempo l'informazione
                  > via ge-0/0/2                             è stata posta in tabella
10.0.8.3/32     *[Local/0] 00:11:43
                  Local
172.16.16.0/21  *[Aggregate/130] 00:36:17
                  Reject
172.16.16.0/24  *[Static/5] 00:06:12 Preference
                  Reject
192.168.5.1/32  *[OSPF/10] 00:00:15, metric 1 ← Costo
                  > via so-0/0/2.900
192.168.10.0/24 *[BGP/170] 00:06:08, localpref 100
                  AS path: 1 I
                  > to 198.51.100.2 via ge-0/1/0.100 ← Interfaccia di uscita
                                                                (Outbound Interface)
. . .           Rete destinazione (in
                  ordine numerico)
                                                                Next-Hop
```

Il processo di *Forwarding*



(*) nel caso comune di interfacce Ethernet

La Tabella di *Forwarding* (FIB): esempio

```
RP/0/5/CPU0:PE4-3# show cef ipv6 2001:db8:51::/64
```

```
. . .
```

Rete destinazione → **2001:db8:51::/64**, version 5, internal 0x14004001 (ptr 0xae8b9e40) [1], 0x0 (0x0), 0x410 (0xaeb4f2f8)

Updated Jul 1 16:13:26.101

Protocol Next-Hop → **via ::ffff:192.168.0.53**, 3 dependencies, recursive [flags 0x6010]

path-idx 0 [0xae6070ac 0x0]

next hop VRF - 'default', table - 0xe0000000

next hop ::ffff:192.168.0.53 via ::ffff:192.168.0.53:0

Next-Hop effettivo → **next hop 172.16.43.2/32 Gi0/7/0/0.41 labels imposed {35 16021}**

↑
Etichette MPLS aggiunte

Forwarding Diretto e Forwarding Indiretto

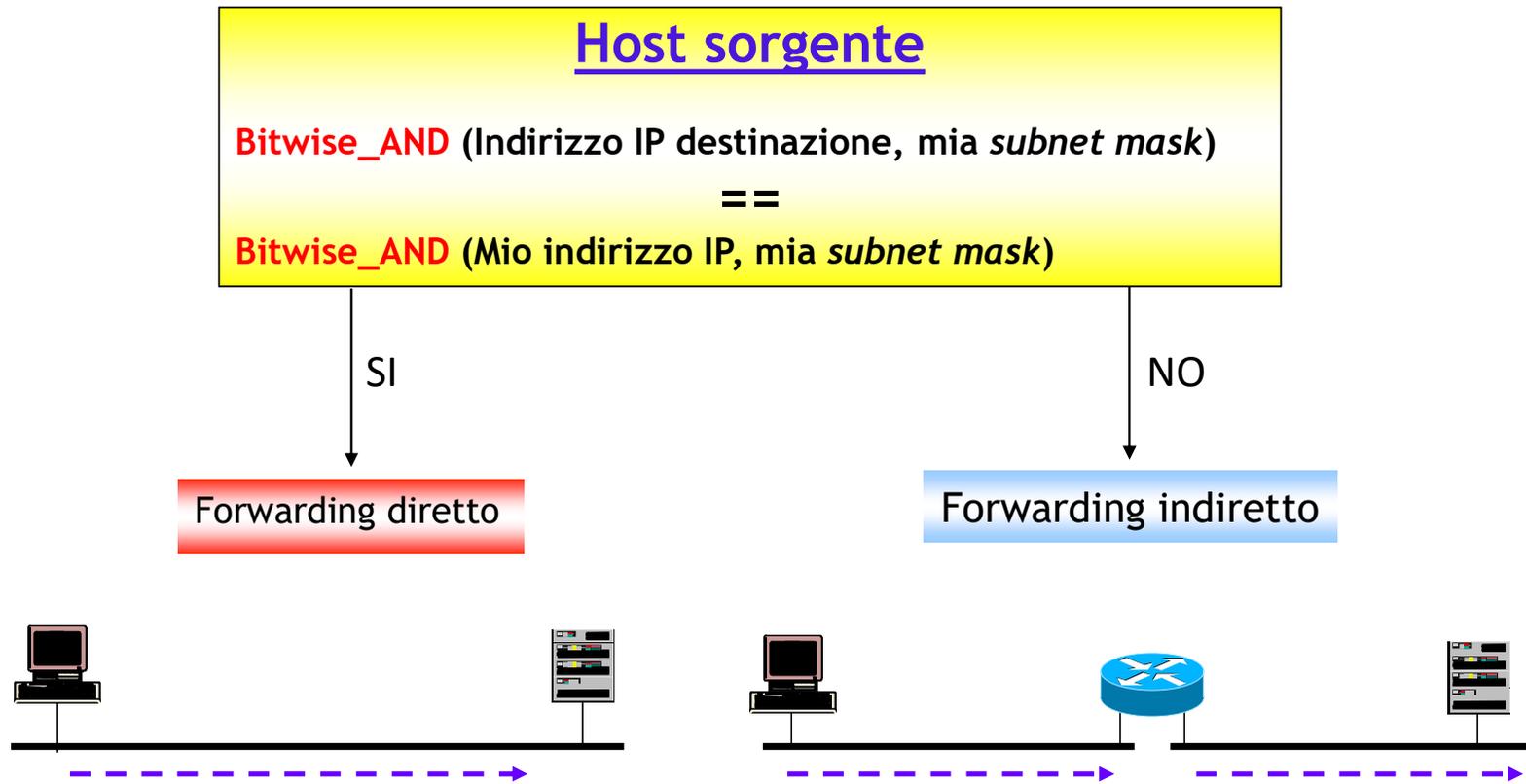
- **Forwarding diretto**

- La trasmissione di un pacchetto IP tra due *host* connessi su una stessa rete IP (stesso prefisso) **non coinvolge i router**
- L'*host* sorgente incapsula il datagramma nella trama L2 e **lo invia direttamente all'*host* destinatario**

- **Forwarding indiretto**

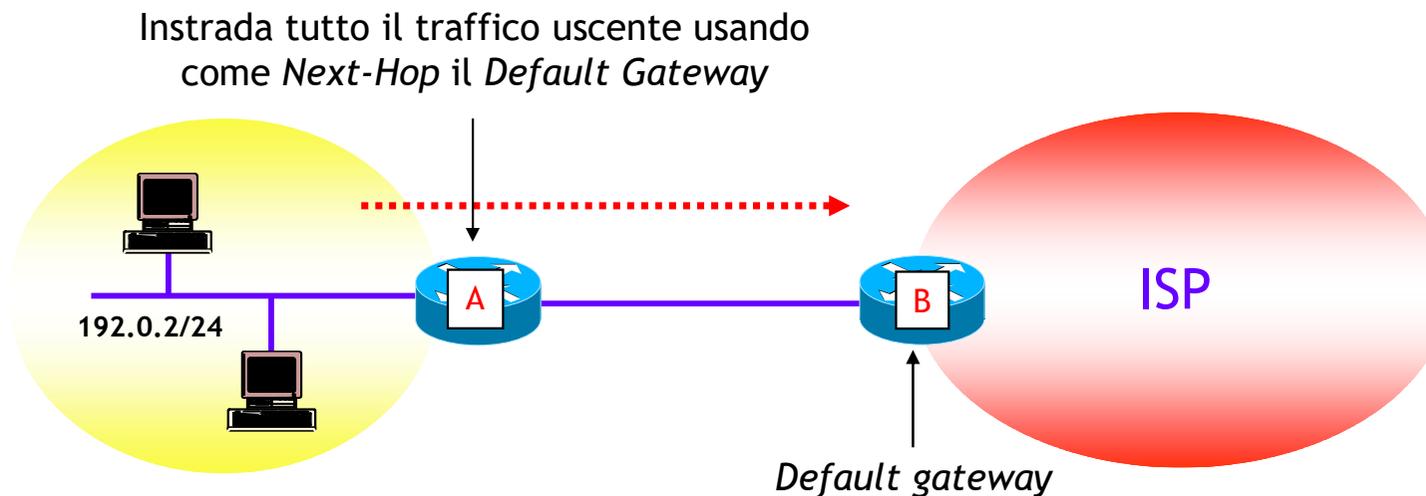
- La trasmissione di un pacchetto IP tra due *host* connessi su differenti reti IP (diverso prefisso) **coinvolge i router**
- L'*host* sorgente incapsula il pacchetto IP nella trama fisica **e lo invia al *default gateway***
- I pacchetti passano da un router all'altro (**hop-by-hop**) finché non raggiungono un router **che può trasmetterli direttamente**

Forwarding Diretto e Forwarding Indiretto (IPv4)



La default route

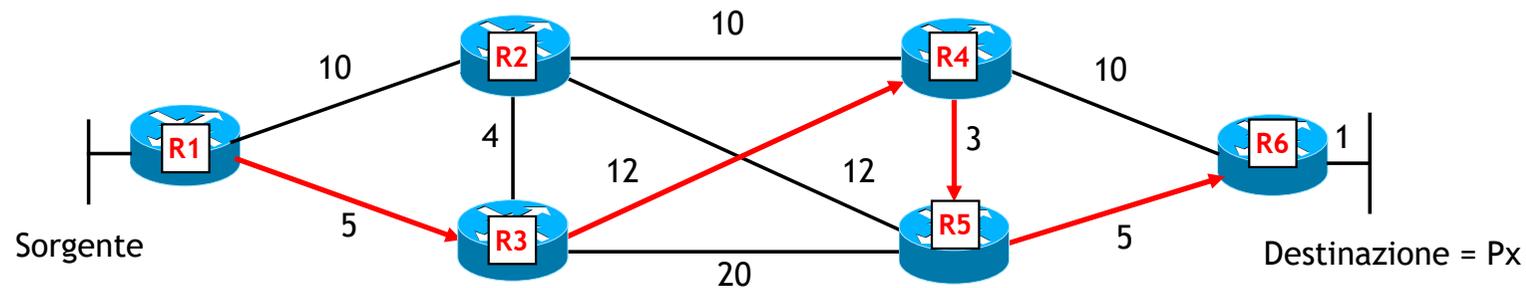
- Serve ad instradare tutti i pacchetti di cui non si ha una conoscenza specifica nella FIB
- Indicata con **0.0.0.0/0** in IPv4 e **::/0** in IPv6
 - Domanda: perché ogni indirizzo IP appartiene alla *default route*?



Come si determina il percorso ottimo?

1. A ciascuna interfaccia viene assegnato un **peso (metrica)**
 - La metrica può essere **assegnata manualmente oppure determinata automaticamente**
 - La determinazione automatica dipende dal tipo di protocollo di routing adottato (es. RIP, OSPF, IS-IS, BGP, ecc.) e dalla particolare tecnologia utilizzata (es. Cisco, Juniper, Nokia, Mikrotik, TIESSE, ecc.)
 - In un collegamento **la metrica può essere asimmetrica** anche se nelle applicazioni pratiche questo si verifica raramente
2. Per ciascun percorso si definisce il **costo** come **somme delle metriche delle interfacce attraversate**
3. Si sceglie come percorso ottimo **quello a minimo costo**
 - Non tutti i protocolli utilizzano questa regola, una eccezione è ad esempio il BGP che ha una sua logica basata su un processo di selezione

Determinazione del percorso ottimo: esempio

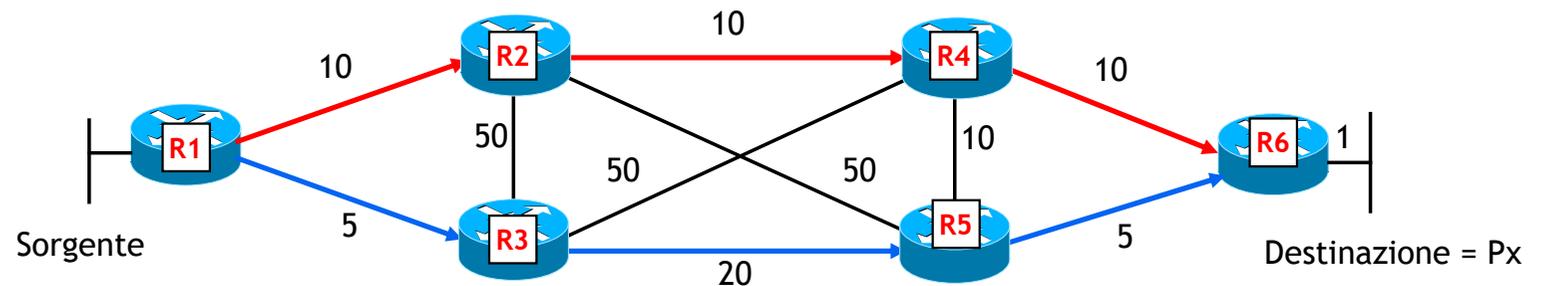


Ipotesi: metriche simmetriche

- Percorso ottimo: R1 --> R3 --> R4 --> R5 --> R6
- Costo: $5 + 12 + 3 + 5 = 25$

Equal Cost MultiPath (ECMP)

- Può accadere che via siano più percorsi a costo minimo (*Equal Cost MultiPath*)
 - In questa situazione è possibile installare più *Next-Hop* nella FIB, che possono essere utilizzati per ripartire il carico sui collegamenti uscenti
 - NOTA: alcuni protocolli prevedono un *Unequal Cost Multipath* (es. EIGRP) o un *load sharing* sulla base della banda dei collegamenti (es. BGP)



Ipotesi: metriche simmetriche

- Percorsi ottimi:
 - R1 --> R2 --> R4 --> R6 (costo = 30)
 - R1 --> R3 --> R5 --> R6 (costo = 30)

Popolazione della RIB

- Le righe della RIB sono
 - Relative a **reti direttamente connesse** (es. segmenti LAN, collegamenti p2p, ecc.)
 - Definite manualmente attraverso **routing statico**
 - Apprese attraverso **protocolli di routing dinamici**
- Routing **statico**: le righe vengono definite manualmente dall'amministratore di rete e **non cambiano finché non vengono modificate manualmente**
- Routing **dinamico**: le righe vengono apprese automaticamente dai router vicini attraverso **protocolli di routing** (es. OSPF, IS-IS, BGP)

Contesa tra protocolli di Routing: il Grado di Preferenza

- PROBLEMA: quale *Next-Hop* installa nella RIB (e quindi nella FIB) un router quando una determinata rete è annunciata da più protocolli di routing dinamici oppure inserita nella RIB staticamente?
 - Esempio: la rete X viene appresa via OSPF che indica come *Next-Hop* ottimo NH1 e anche via BGP che indica come *Next-Hop* ottimo NH2
- La contesa si risolve attraverso un **Grado di Preferenza** assegnato a ciascun protocollo
 - Il Grado di Preferenza assume denominazioni diverse nelle varie tecnologie (es. Cisco la indica come *Distanza Amministrativa*, il JunOS come *Route Preference*, Mikrotik come *Route Distance*, ecc.
 - Ogni costruttore utilizza valori di default diversi per ciascun protocollo, valori che sono locali al router e che possono essere variati via configurazione
- Vince il protocollo con il **Grado di Preferenza più basso**

Il Grado di Preferenza nei router Cisco (Distanza Amministrativa)

- Alcuni valori significativi di Distanza Amministrativa nei router Cisco

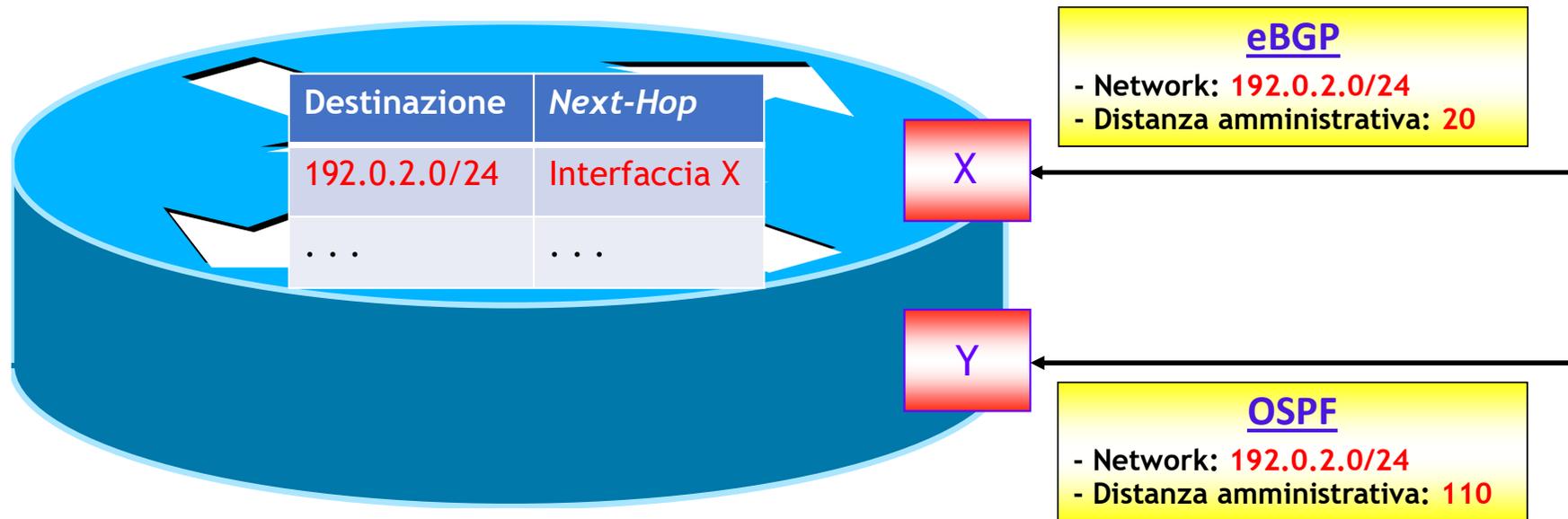
Sorgente di routing	Distanza Amministrativa
Rete direttamente connessa	0
Route statica (Next-Hop Interf.)	0
Route statica (Next-Hop IP)	1
External BGP	20
OSPF	110
IS-IS	115
RIPv1/v2	120
Internal BGP	200
Protocolli sconosciuti	255

Il Grado di Preferenza nei router Juniper

- Alcuni valori significativi di *Route Preference* nei router Juniper

Sorgente di routing	Distanza Amministrativa
Rete direttamente connessa	0
Route statica	5
OSPF (route interne)	10
IS-IS Livello 1 (route interne)	15
IS-IS Livello 2 (route interne)	18
RIPv1/v2	100
OSPF (route esterne)	150
IS-IS Livello 1 (route esterne)	160
IS-IS Livello 2 (route esterne)	165
BGP (internal/external)	170

Il Grado di Preferenza: esempio in un router Cisco



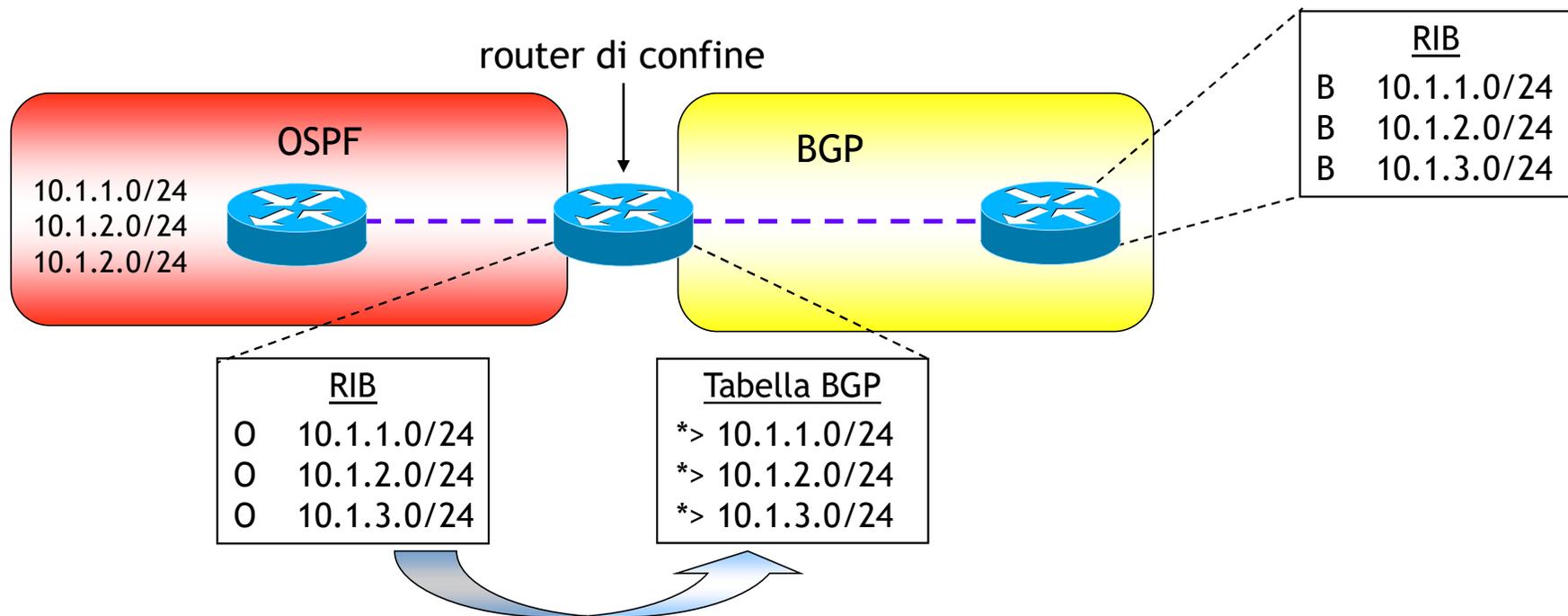
- NOTA: in un router Juniper con lo stesso scenario la preferenza verrebbe data al *Next-Hop OSPF* (interfaccia Y)

Convergenza dei protocolli di routing

- La convergenza è il processo di **notifica e allineamento di tutti i router** di un AS a fronte di un cambio di topologia (es. per fuori servizio di un collegamento o nodo)
- Il tempo che intercorre tra l'istante in cui avviene una variazione nella topologia della rete e quello in cui tutti i router hanno aggiornato le proprie RIB viene definito **tempo di convergenza**
- Quando un evento sulla rete rende un percorso ottimo indisponibile, il tempo di convergenza **deve essere il più basso possibile**
- Tre variabili giocano un ruolo fondamentale
 1. riscontrare il cambiamento
 2. diffondere le informazioni dell'evento
 3. adattarsi al cambiamento

Redistribuzione di informazioni di routing

- La redistribuzione è il processo che consente di **trasferire informazioni di routing tra domini di routing che utilizzano diversi protocolli di routing**
- La funzione di redistribuzione è svolta da un **router di confine** che implementa i due diversi protocolli presenti nei due domini di routing



Di cosa parlerò ...

#1

A cosa serve il routing IP

#2

Concetti fondamentali

#3

Routing statico e dinamico

#4

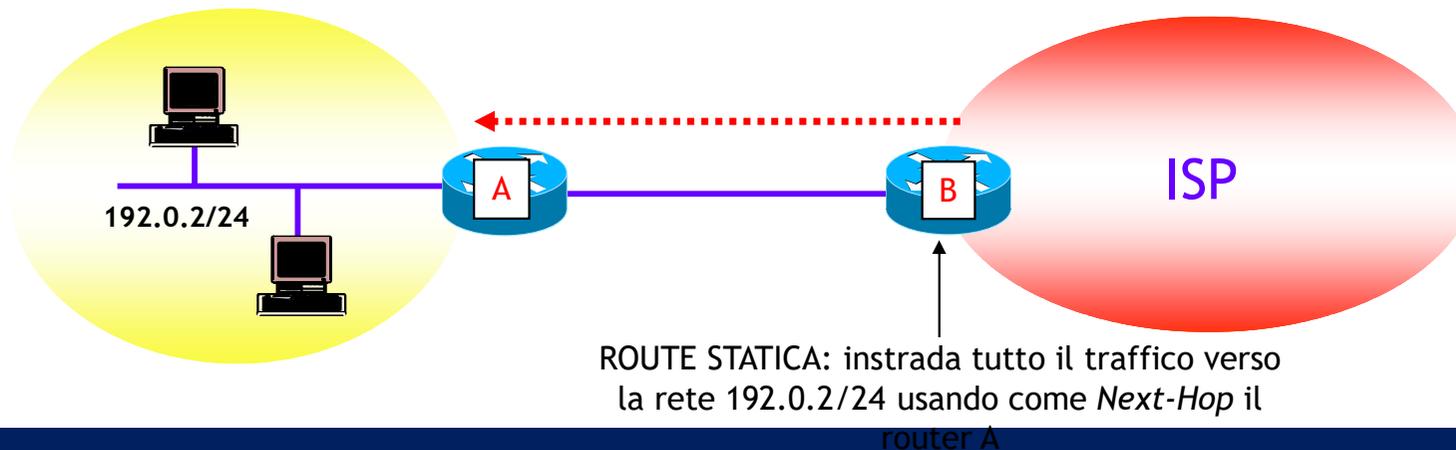
Protocolli di routing *Distance/Path Vector* e *Link State*

#5

Routing *multicast*

Route statiche

- Sono percorsi definiti **manualmente**
 - Le informazioni minimali da specificare in fase di configurazione sono il **prefisso destinazione** e il **Next-Hop**
 - È possibile specificare altre opzioni come ad esempio Grado di Preferenza, metriche, ecc.
- Molto utili in situazioni particolari come ad esempio nel caso di **collegamento singolo** tra un cliente e un *Service Provider*
- **NOTA BENE:** le route statiche sono **unidirezionali**



Route statiche nella RIB di un router Juniper

```
tt@router> show route table inet.0 protocol static
```

```
inet.0: 13 destinations, 15 routes (13 active, 0 holddown, 0 hidden)  
+ = Active Route, - = Last Active, * = Both
```

```
192.0.2.0/24      *[Static/5] 00:03:58  
                  > to 192.0.2.1 via ge-0/0/0.0  
203.0.113.0/24   *[Static/5] 00:01:14  
                  > to 192.0.2.1 via ge-0/0/0.0  
191.58.100.0/24  *[Static/5] 00:01:14  
                  > to 192.0.2.1 via ge-0/0/0.0
```

Grado di Preferenza
(5 = default)

Rete Destinazione

Next-Hop

Interfaccia di uscita

Configurazione di route statiche nei router Cisco

'prefisso lunghezza-prefisso' per route statiche IPv4

- IOS e IOS XE



```
router(config)# {ip | ipv6} route prefisso/lunghezza-prefisso {IP-Next-Hop  
| interfaccia} [distanza amministrativa] [description text] [tag tag]  
[permanent]
```

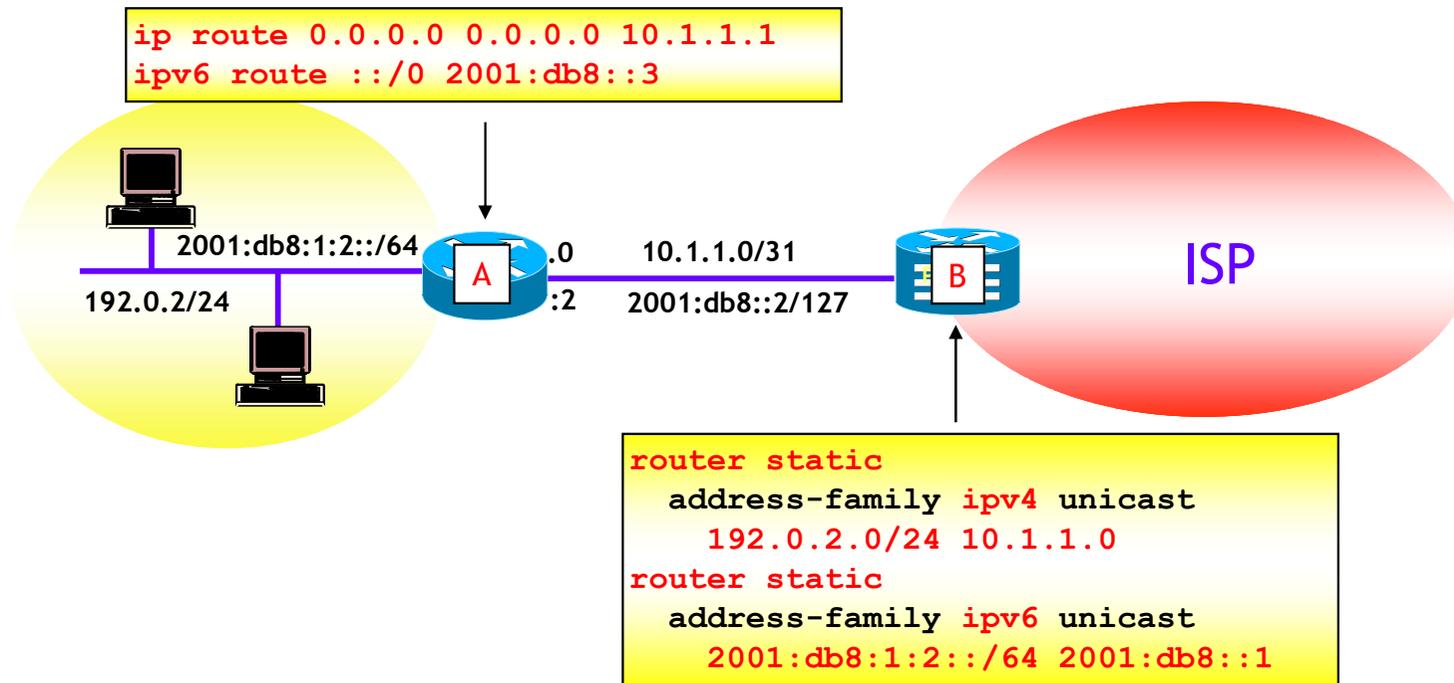
- IOS XR

```
router static  
address-family {ipv4 | ipv6} unicast  
prefix/maschera {IP-next-hop | interfaccia} [distanza amministrativa]  
[description text] [tag tag] [permanent]
```

- NOTA: l'opzione '**permanent**' consente di non rimuovere la route statica dalla RIB anche nel caso in cui il *Next-Hop* diventi irraggiungibile
 - Utile per evitare *route flapping*

Route statiche nei router Cisco: esempio

- Configurazione di route statiche IPv4/IPv6
 - Sul router A una *default route* statica con *Next-Hop* il router B
 - Sul router B una route statica verso il prefisso LAN con *Next-Hop* il router A



Configurazione di route statiche nei router Juniper

Da inserire solo nel caso
di route statiche IPv6

```
[edit routing-options < rib inet6.0 >]
tt@router# show
static {
  defaults {
    opzioni;
  }
  route prefisso-destinazione {
    next-hop next-hop;
# OPPURE
    qualified-next-hop next-hop {
      preference grado-di-preferenza;
    }
    opzioni;
  }
}
```

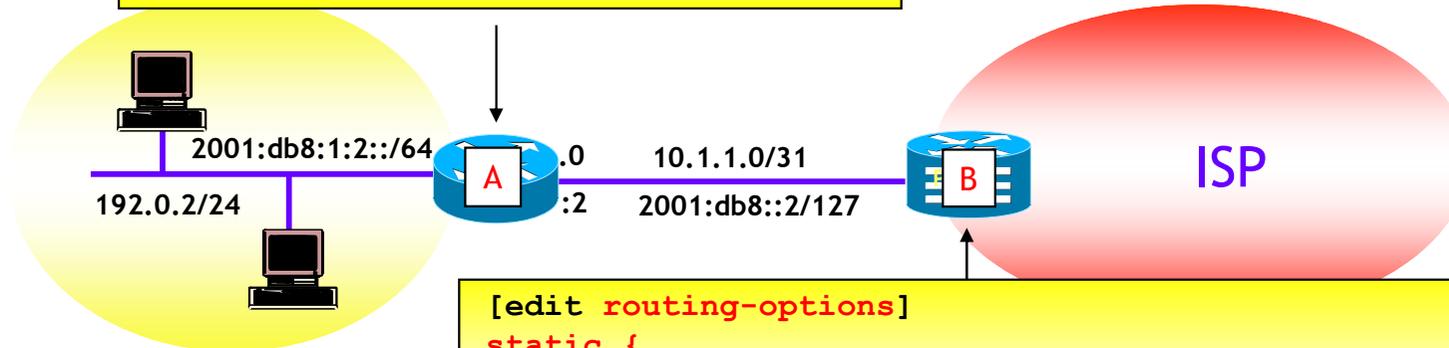
Opzioni valide per tutte le
route statiche configurate

Opzioni valide per la particolare
route statica configurata

Route statiche nei router Juniper: esempio

```
[edit routing-options]
static {
  route 0.0.0.0/0 next-hop 10.1.1.1;
}

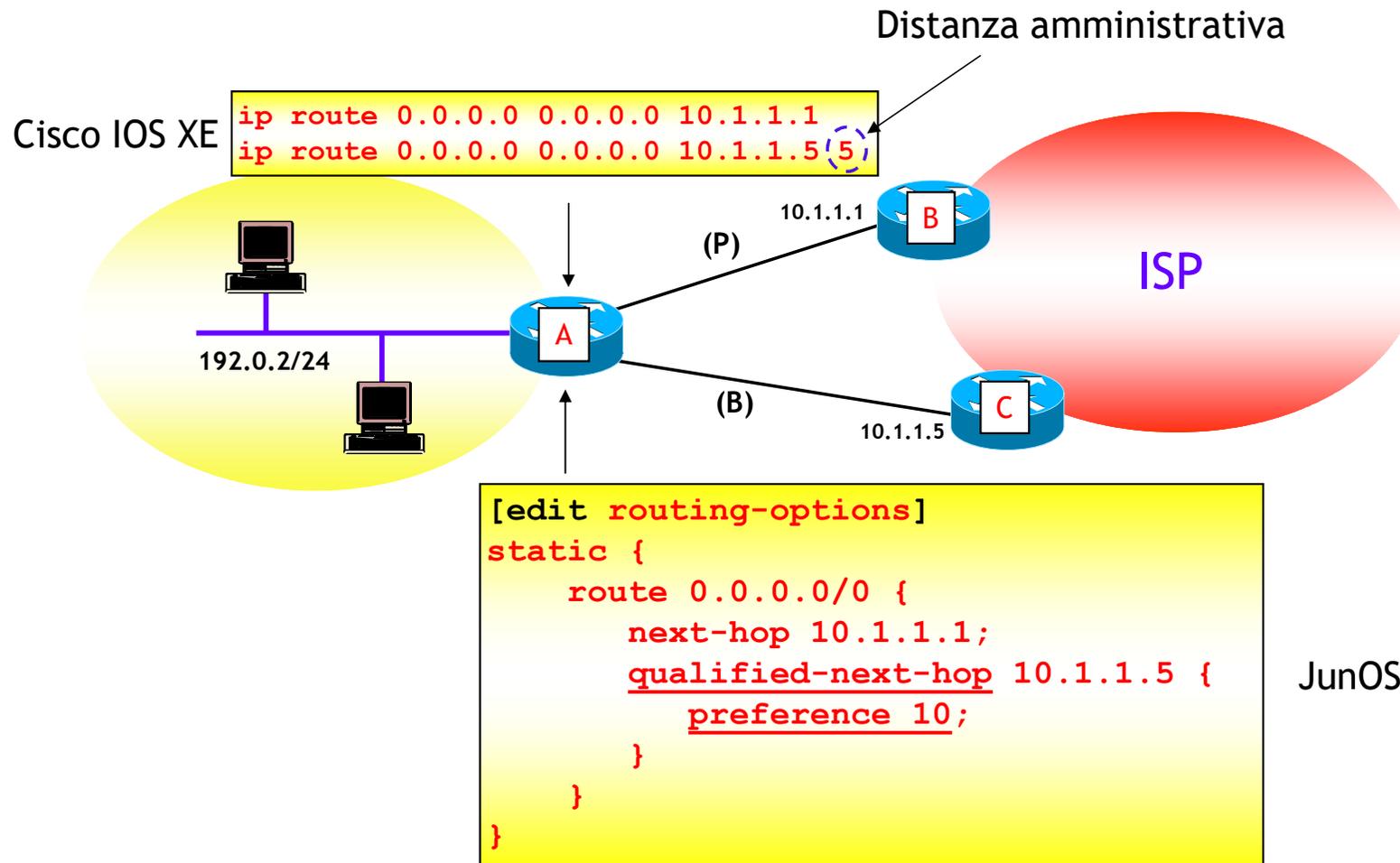
[edit routing-options rib inet6.0]
static {
  route ::/0 next-hop 2001:db8::3;
}
```



```
[edit routing-options]
static {
  route 192.0.2.0/0 next-hop 10.1.1.0;
}

[edit routing-options rib inet6.0]
static {
  route 2001:db8:1:2::/64 next-hop 2001:db8::2;
}
```

Floating Static Route



Una generalizzazione del routing statico: il PBR

- Il *Policy Based Routing* (PBR) è un meccanismo che consente di alterare le decisioni di routing presenti nella RIB
 - Viene applicato a tutti i pacchetti in ingresso ad una interfaccia dove questo è abilitato
 - Può essere pensato come una generalizzazione delle route statiche, dove il *Next-Hop* viene scelto non solo sulla base di un prefisso destinazione, ma anche di altri attributi (es. indirizzo IP sorgente, tipo di protocollo di trasporto, porte, ecc.)
 - NOTA: nel gergo del JunOS il PBR viene dicato come FBF (*Filter Based Forwarding*)
- In presenza di PBR, l'inoltro dei pacchetti, che è normalmente basato sull'indirizzo di destinazione, viene gestito come segue:
 - Se il pacchetto è parte di un insieme predefinito (via configurazione) viene inoltrato in base ad un *Next-Hop* definito manualmente dall'amministratore di rete
 - Altrimenti viene instradato normalmente secondo le informazioni presenti nella FIB
 - NOTA: se il *Next-Hop* definito manualmente non funziona (ad esempio, il *Next-Hop* è irraggiungibile, l'interfaccia *Next-Hop* è down), il PBR utilizza la FIB

Routing dinamico

- Con il routing dinamico le righe della RIB vengono **costruite e aggiornate dinamicamente** attraverso dei **protocolli di routing**, senza alcun intervento manuale
- Un protocollo di routing descrive
 - Il **formato dei messaggi** scambiati tra i router
 - Le procedure utilizzate per **scoprire la topologia della rete**
 - Le procedure utilizzate per **la determinazione dei cammini ottimi**
 - Le azioni messe in atto per **aggiornare automaticamente le RIB in caso di cambio della topologia della rete**
- Esistono due categorie di protocolli di routing dinamici (vedi prossima sezione)
 - *Distance/Path Vector*
 - *Link State*
- Tutti i protocolli di routing sono costruiti su un algoritmo adattativo che, attraverso opportune operazioni, propone il percorso migliore
 - *Distance Vector*: algoritmo di **Bellmann-Ford**
 - *Link State*: algoritmo di **Dijkstra**
 - *Path Vector*: **processo di selezione**

Routing statico o dinamico?

- **Statico**

- Ideale in connessioni tra una rete *stub* ed un ISP
- Non impatta sulle risorse della CPU del router
- Utile per la gestione dei *backup*
- Se usata come unica tipologia di routing **non consente la realizzazione di reti scalabili**

- **Dinamico**

- Al cambiamento della topologia **ricalcola** il nuovo percorso verso la destinazione analizzando gli aggiornamenti di routing ricevuti
- Garantisce, ove configurato in maniera gerarchica, la **scalabilità** della rete
- Una errata implementazione può pregiudicare le prestazioni di rete
 - Possono verificarsi problemi di loop

- **Best practice**

- Utilizzare il routing statico solo in situazioni particolari (es. collegamento di un router con un solo collegamento o al massimo due verso lo stesso ISP)
- Per tutto il resto utilizzare il routing dinamico

Di cosa parlerò ...

#1

A cosa serve il routing IP

#2

Concetti fondamentali

#3

Routing statico

#4

Protocolli di routing *Distance/Path Vector* e *Link State*

#5

Routing *multicast*

Tipologie di protocolli

Approccio *Distance Vector*

- - nato con ArpaNet - -

- Ciascun router mantiene una tabella che contiene, per ogni destinazione a lui (al momento) nota, la **distanza**, e il **Next-Hop** per raggiungerla
- Periodicamente, ogni router invia la propria tabella **solo ai vicini diretti**
- Usando le informazioni ricevute **dai vicini**, ogni router aggiorna la propria tabella

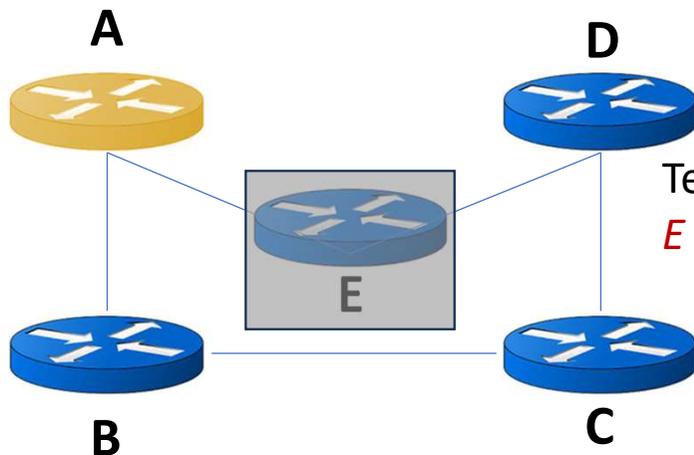
Approccio *Link State*

- - più recente - -

- Ciascun router mantiene una *database* che descrive la topologia della rete (al momento) nota, **includere le metriche associate a ogni interfaccia**
- Tramite un algoritmo ottimizzato ciascun router determina il percorso ottimo verso ciascuna destinazione nota
- Il *Next-Hop* del percorso ottimo viene inserito nella RIB e quindi nella FIB

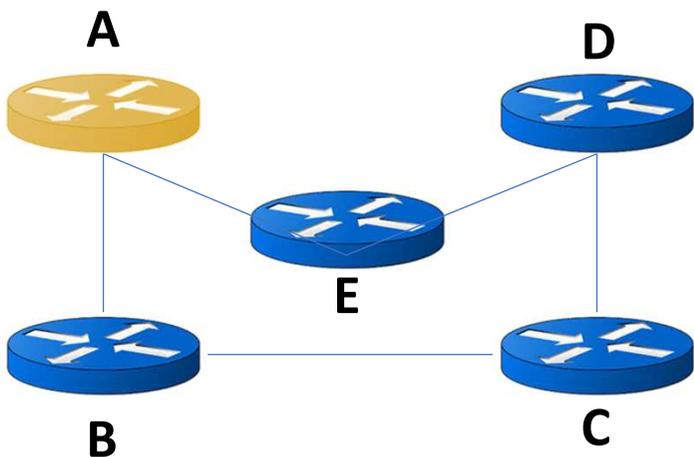
Distance Vector - Esempio di funzionamento

...se A vuol mandare pacchetti a D...



Tempo t ; info da E non disponibili

Routing Table del Router A		
Dest.	Next Hop	Dist.
A	-	0
B	b	1
C	b	2
D	b	3
E	?	?

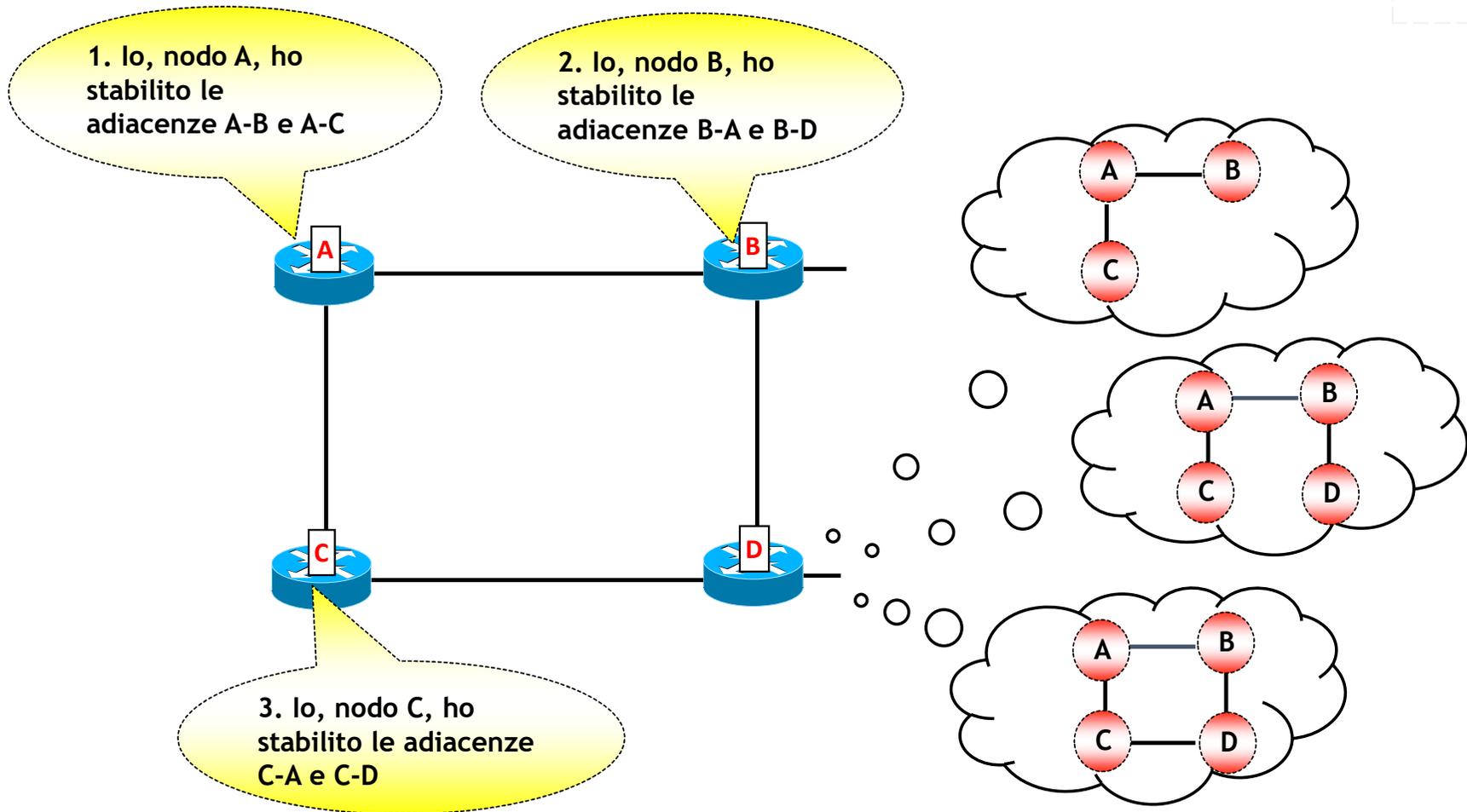


Tempo $t+\Delta$; sono arrivate le info da E

A «scopre» un percorso migliore verso D!

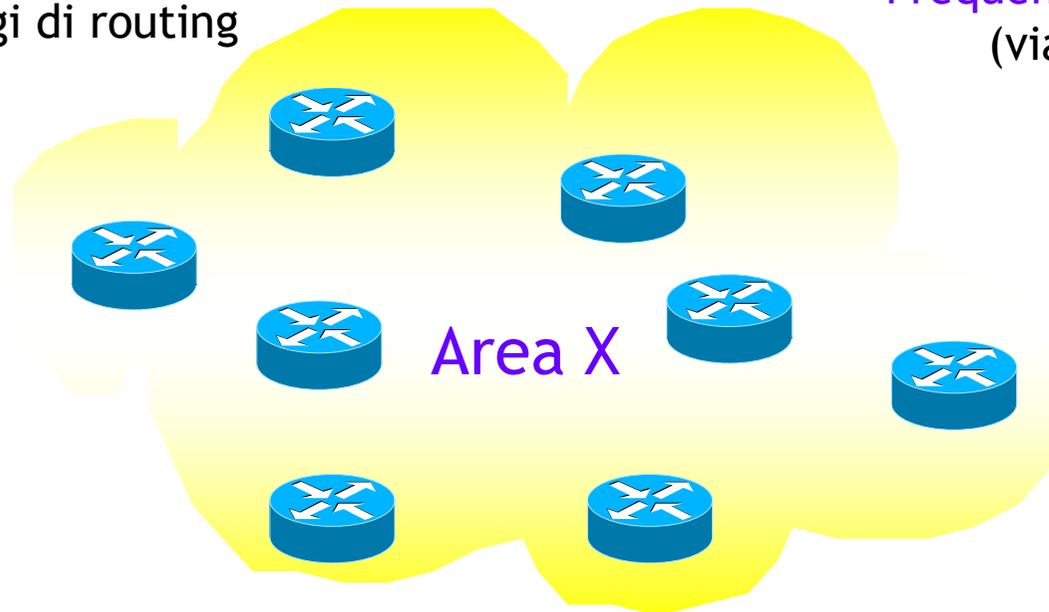
Routing Table del Router A		
Dest.	Next Hop	Dist.
A	-	0
B	b	1
C	b	2
D	e	2
E	e	1

Principio del *Link State*



Protocolli *Link State*: scalabilità e stabilità

Consumo eccessivo di banda a causa del *flooding* dei messaggi di routing



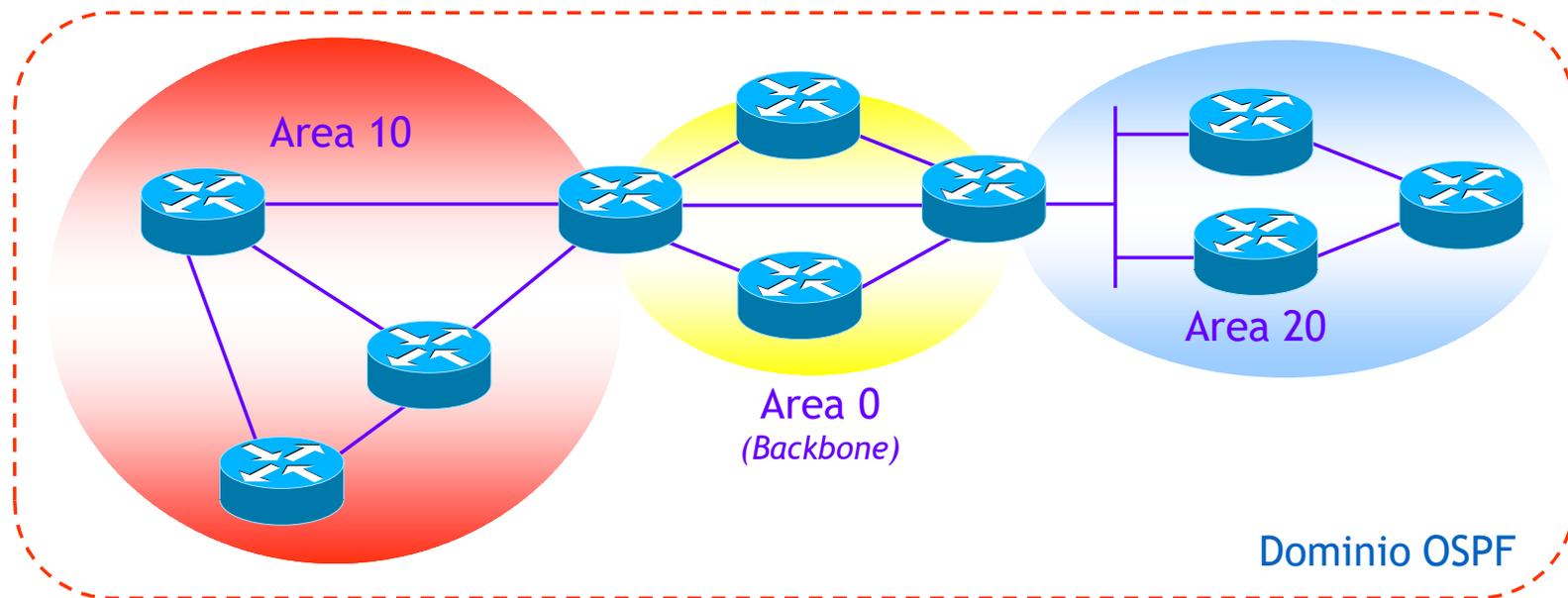
Frequenti ricalcoli dei percorsi ottimi (via SPF) pesanti per la CPU

RIB grandi e sensibili alle variazioni

Consumo eccessivo di memoria a causa di LSDB molto grandi

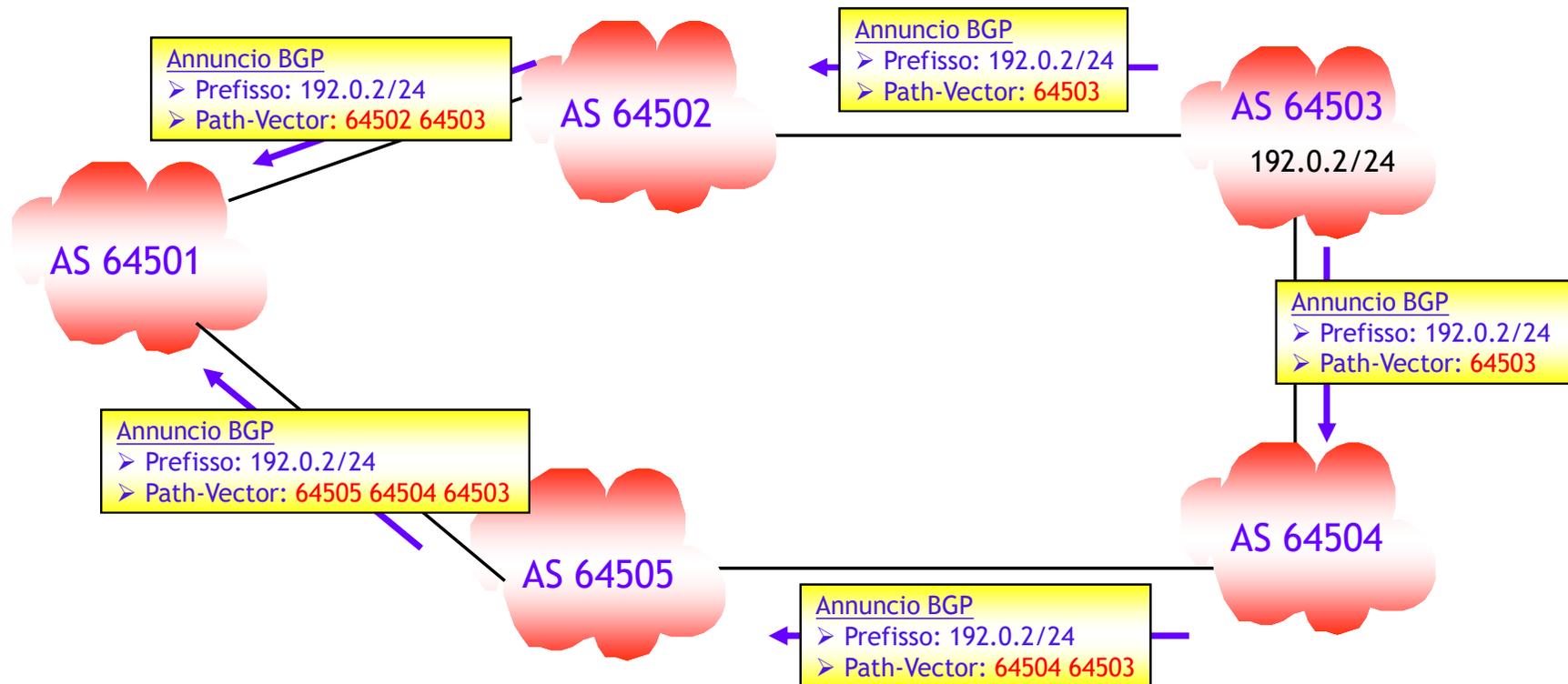
Protocolli *Link State*: routing gerarchico

- In reti di grandi dimensioni, per motivi di **scalabilità** e soprattutto di **stabilità** il dominio di routing viene **partizionato in entità indipendenti** (aree) **interconnesse attraverso un livello superiore** (routing gerarchico)
 - OSPF: area 0 (area *backbone*)
 - IS-IS: topologia di Livello 2



E ci sarebbe anche un protocollo *Path Vector*: il BGP ...

- Il BGP è un protocollo di tipo *Path Vector*
 - Ogni annuncio BGP contiene (anche) la lista di tutti gli AS attraversati (*Path Vector*)



Principali protocolli di routing *unicast*

	Algoritmo di ricerca del percorso ottimo	Protocollo
<i>Distance Vector</i>	Bellmann-Ford	RIPv1/v2, RIPv6
<i>Distance Vector</i> avanzato	DUAL	EIGRP (*)
<i>Link State</i>	SPF (Dijkstra)	OSPFv2/v3, IS-IS
<i>Path Vector</i>	Processo di selezione	BGP

(*) protocollo proprietario Cisco (successivamente standardizzato nella RFC 7868 - maggio 2016)

Di cosa parlerò ...

#1

A cosa serve il routing IP

#2

Concetti fondamentali

#3

Routing statico

#4

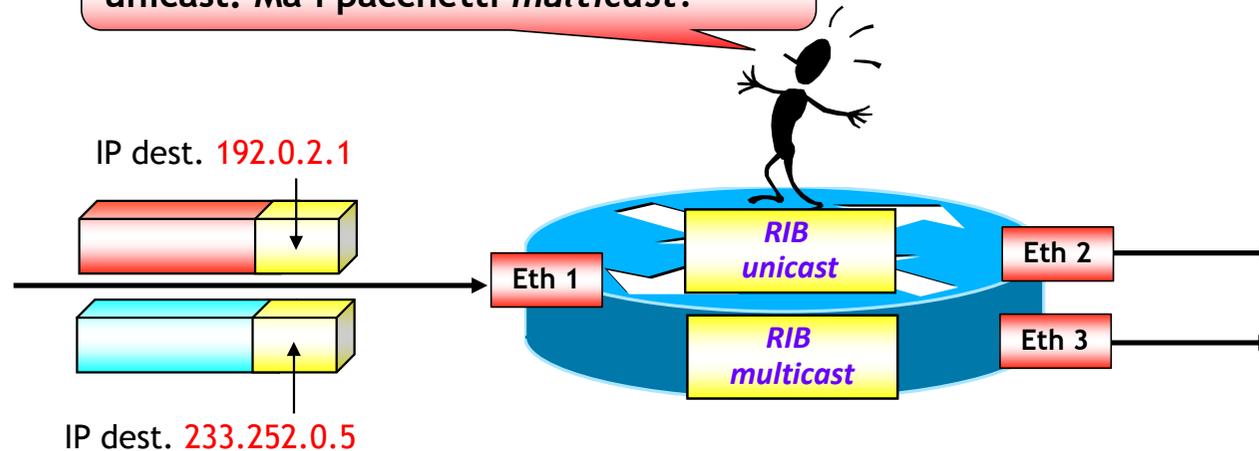
Protocolli di routing *Distance/Path Vector* e *Link State*

#5

Routing multicast

Routing *unicast* vs *multicast*

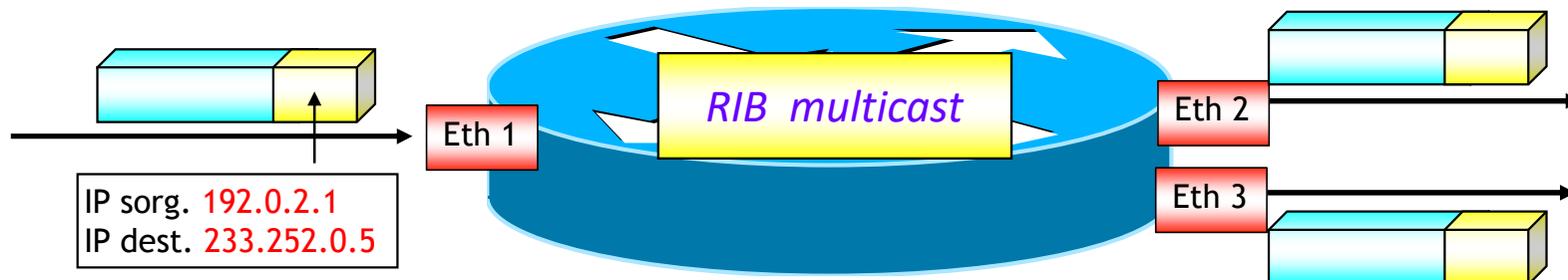
Io sò bene come instradare i pacchetti unicast. Ma i pacchetti *multicast*?



- Il problema principale del routing multicast è che **l'indirizzo destinazione rappresenta un gruppo dinamico di destinatari**
 - Come fa il router a sapere **dove si trovano i destinatari?**
 - A quali interfacce il router deve inoltrare i pacchetti con indirizzo IP destinazione multicast?
 - Come può costruire i **percorsi ottimi?**

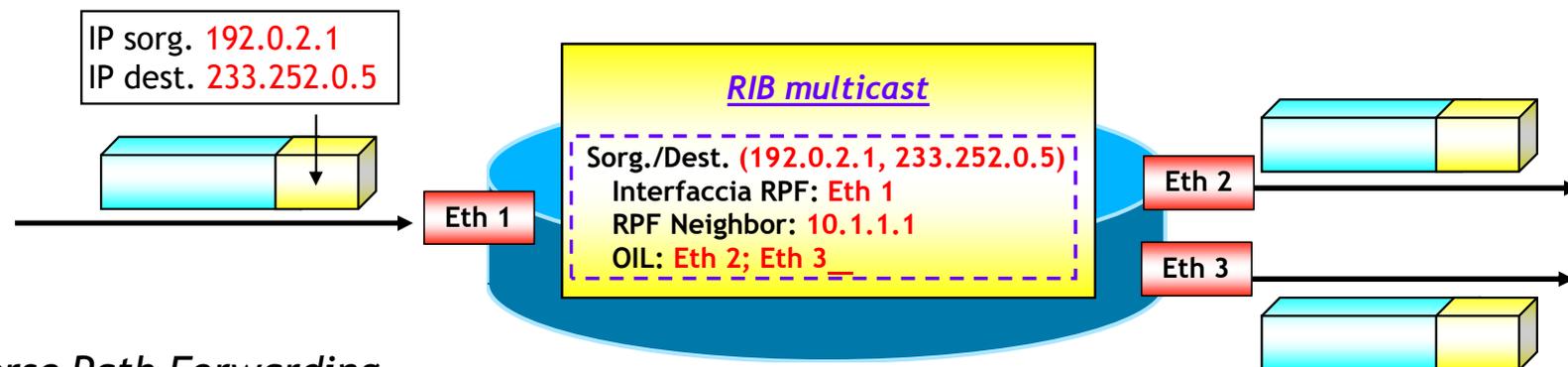
Router *multicast*

- Come per qualsiasi altro router, le **funzioni fondamentali di un router multicast** sono due
 - Creazione e mantenimento delle **tabelle di routing** utilizzate dai pacchetti multicast
 - **Commutazione (*forwarding*)** dei pacchetti multicast
- La commutazione dei pacchetti *multicast*, a differenza della commutazione dei pacchetti *unicast*, richiede in generale che **il pacchetto venga inoltrato su più di una interfaccia di uscita**



RIB multicast

- Il generico elemento di una RIB multicast è costituito da **tre componenti fondamentali**
 - La coppia <*Sorgente*, *Destinazione*>
 - *Sorgente* = indirizzo IP della sorgente del traffico multicast
 - *Destinazione* = gruppo multicast destinazione
 - L'interfaccia per il **controllo RPF** e l'*RPF Neighbor*
 - L'insieme delle **interfacce di uscita** su cui inoltrare il pacchetto multicast (**OIL**, *Outgoing Interface List*)



(*) RPF=Reverse Path Forwarding

Protocolli di routing *multicast*

- Sono lo strumento per la **costruzione e il mantenimento della RIB *multicast***
- Funzioni principali
 - Determinare, per ciascuna sorgente di traffico, l'interfaccia *upstream*
 - Determinare, per ciascuna coppia <Sorgente, Gruppo multicast>, l'insieme delle interfacce *downstream*
 - Costruire e gestire gli **alberi *multicast*** per un instradamento ottimale del traffico
- Tre classi fondamentali
 - Protocolli *Dense mode* (DVMRP, PIM-DM): utilizzano un modello di tipo *push*: assumono che su ciascun router della rete siano presenti ricevitori di traffico *multicast*
 - Protocolli *Sparse mode* (CBT, PIM-SM): utilizzano un modello di tipo *pull*: assumono che il traffico *multicast* non sia voluto se non esplicitamente richiesto
 - Protocolli *Link-State* (MOSPF): non più utilizzati

Alberi *multicast*

- Gli alberi multicast sono utilizzati per **descrivere i percorsi utilizzati** da una rete per inoltrare traffico multicast in modo ottimo a tutti i ricevitori interessati
 - **Nodi**: sono i router dove viene diramato il traffico
 - il **nodo radice** è il router da dove si dipana l'albero multicast
 - I nodi dove sono **attestate le sorgenti** di traffico multicast sono detti *first-hop router*
 - **Foglie (*last-hop router*)**: sono i router dove è attestato almeno un ricevitore
- Due tipi:
 - Alberi **sorgente (*Source Tree*)**, spesso indicati con *SPT (*Shortest Path Tree*)*
 - Sono alberi che hanno come nodo radice un *first-hop router*
 - Alberi **condivisi (*Shared Tree*)**, spesso indicati con *RPT (*Rendezvous-Point Tree*, o anche *Root Path Tree*)*
 - Sono alberi che hanno come nodo radice un router al quale converge il traffico di più sorgenti
 - Il nodo radice viene detto *Rendezvous-Point*

Ultima Diapositiva (finalmente ...)



Grazie per l'attenzione