

# Routing

# Introduzione

- Il livello 3 della pila ethernet ha il compito di muovere i pacchetti dalla sorgente attraversando più sistemi
- Il livello di network deve quindi:
  - Scegliere di volta in volta il cammino migliore
  - Gestire il flusso dei dati
  - Gestire le congestioni
  - Gestire le problematiche relative dovute alla presenza di più reti
- Il routing permette la comunicazione tra due nodi differenti anche se non sono collegati direttamente

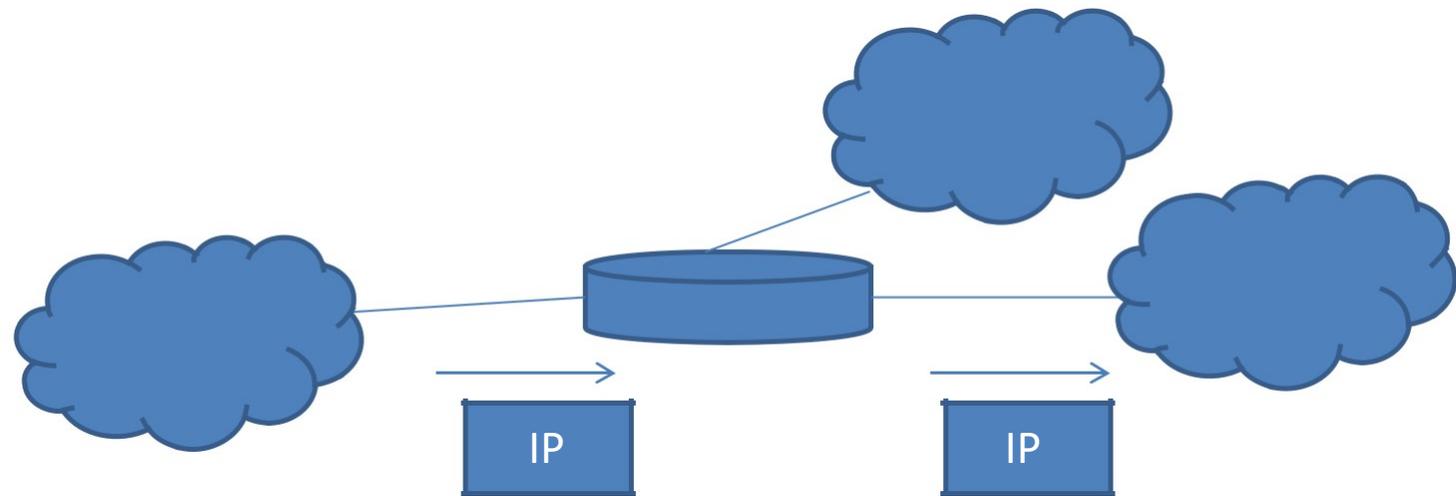
# Routing

- Al livello 3, si effettua la commutazione verso un Service Access Point (SAP) in base all'indirizzo posto sul pacchetto dati e, in base ad una tabella di instradamento che contiene le informazioni sul cammino
- Esistono due tipi di dispositivi:
  - Terminali della rete
  - Nodi di commutazione

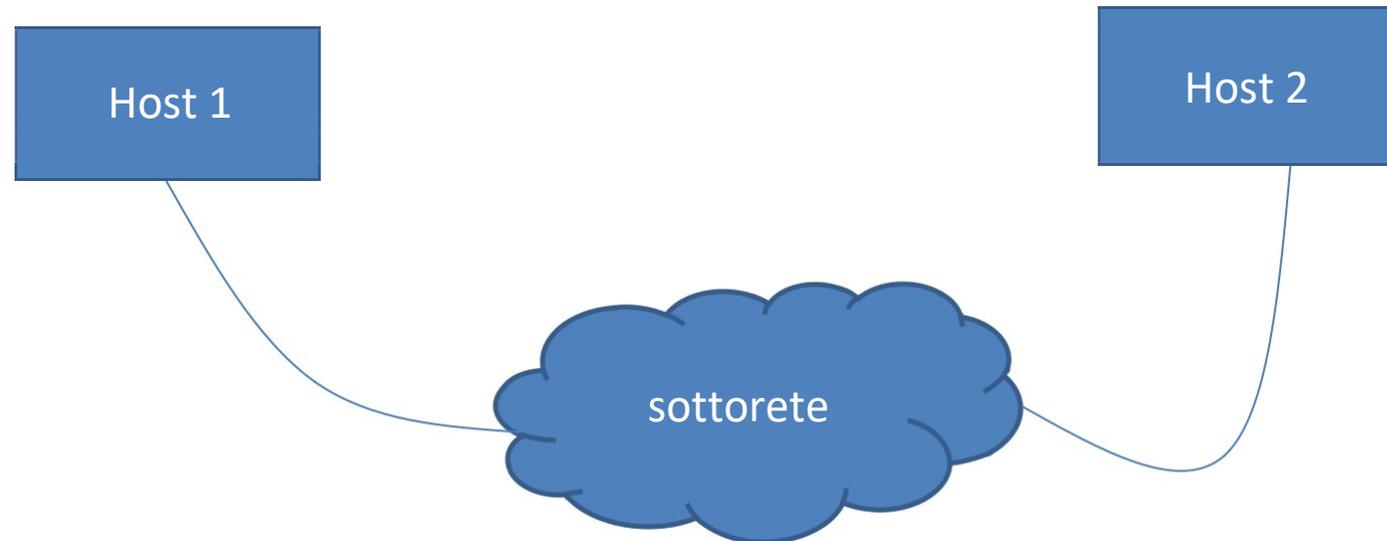
# Instradamento

- I router hanno il compito di ricevere pacchetti dati, analizzarli ed instradarli verso gli indirizzi indicati
- Esistono due tipi di instradamento:
  - Instradamento diretto: la trasmissione avviene tra due host connessi con una singola rete logica e non coinvolge router intermedi. Il trasmettitore risolve l'indirizzo fisico MAC del destinatario tramite il protocollo ARP
  - Instradamento indiretto: l'host di destinazione non è connesso direttamente alla rete alla quale appartiene l'host. L'unica possibilità è quella di passare attraverso più router. Il mittente fornisce l'indirizzo del prossimo router next hop oppure, l'indirizzo del default gateway.

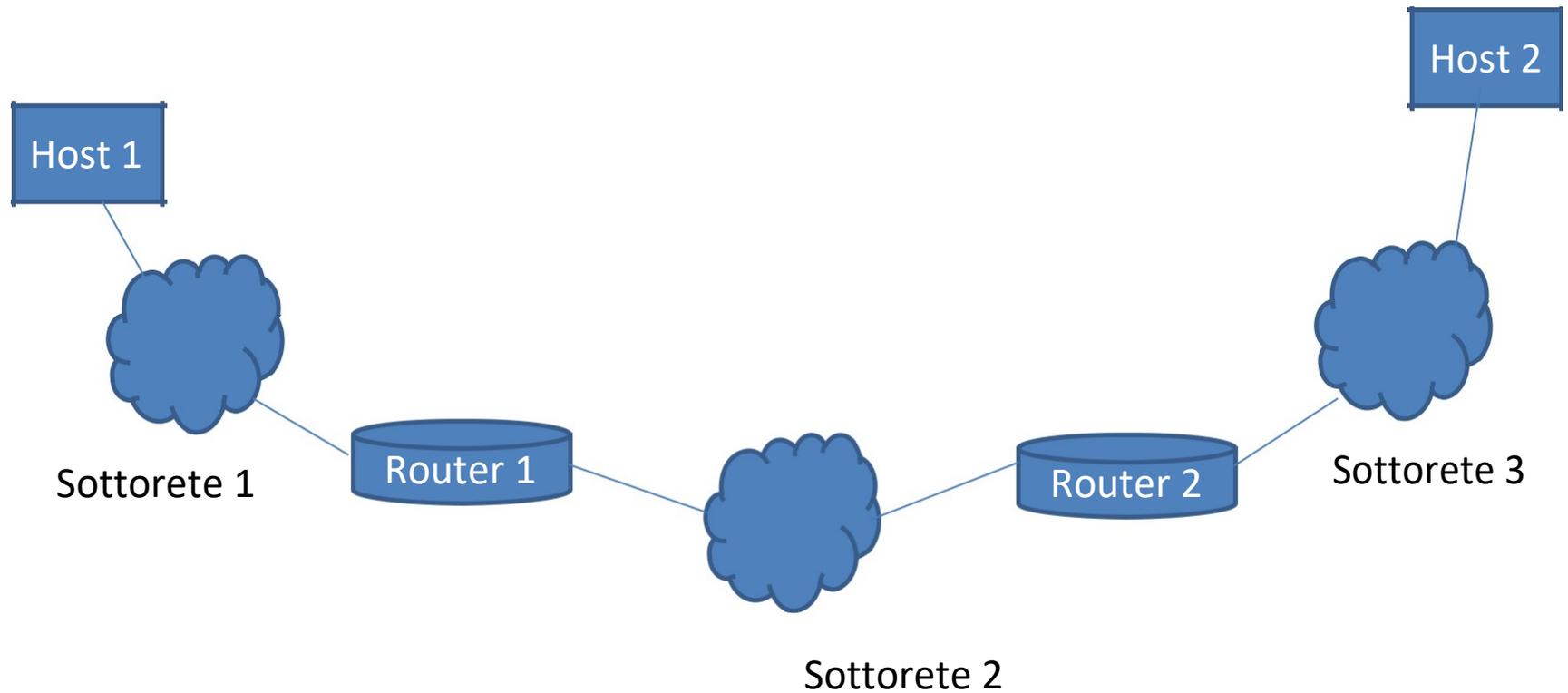
# Instradamento



# Instradamento diretto



# Instradamento indiretto

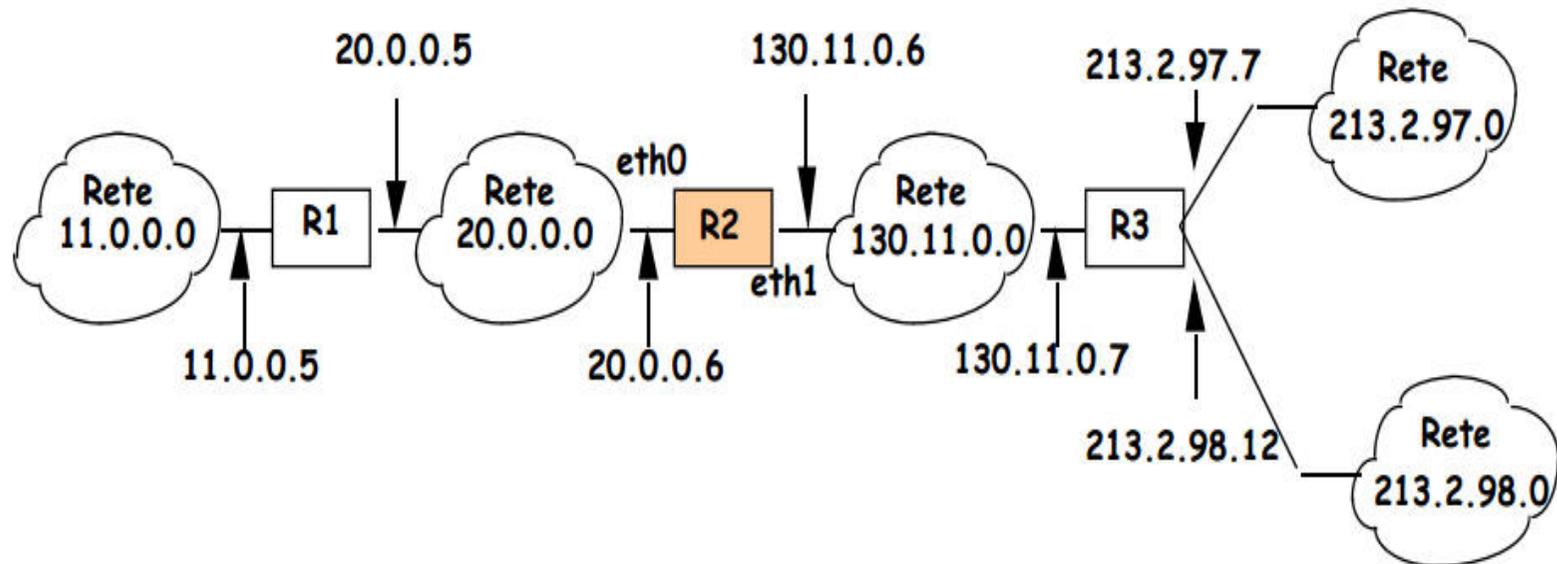


# Tabella di instradamento

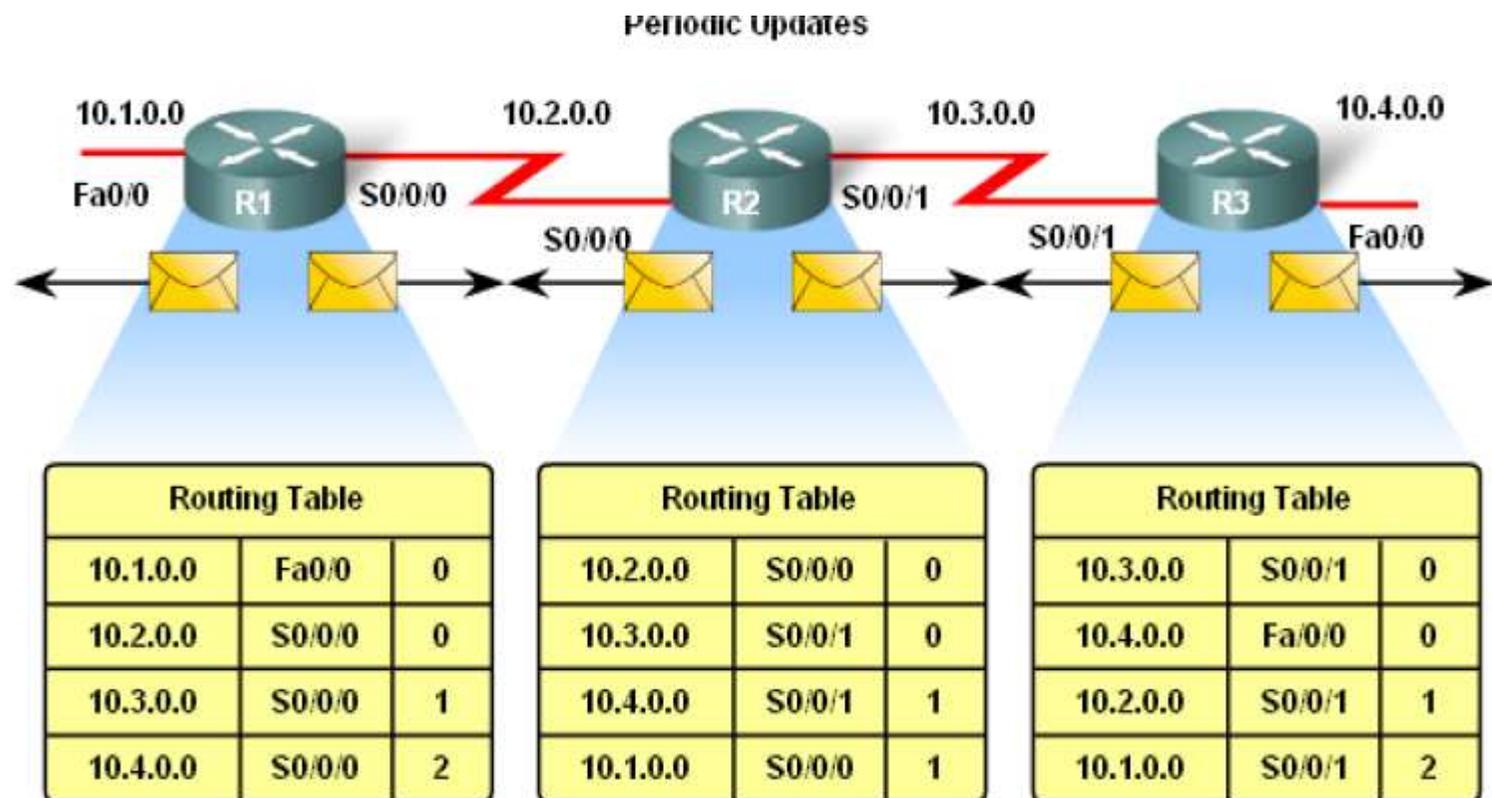
- L'instradamento IP si basa su tabelle presenti su host e router che prendono il nome di tabelle di instradamento o routing.
- Nelle tabelle di instradamento è elencato per ciascuna sottorete, il relativo netID e l'indirizzo del router di inoltra
- Una tabella di instradamento contiene un insieme di righe che contiene i seguenti quattro indirizzi:
  - Indirizzo della rete di destinazione
  - Maschera di rete
  - Interfaccia su cui inoltrare
  - Indirizzo del next hop
- Ad ogni percorso è associato il costo per raggiungere la destinazione

# Instradamento

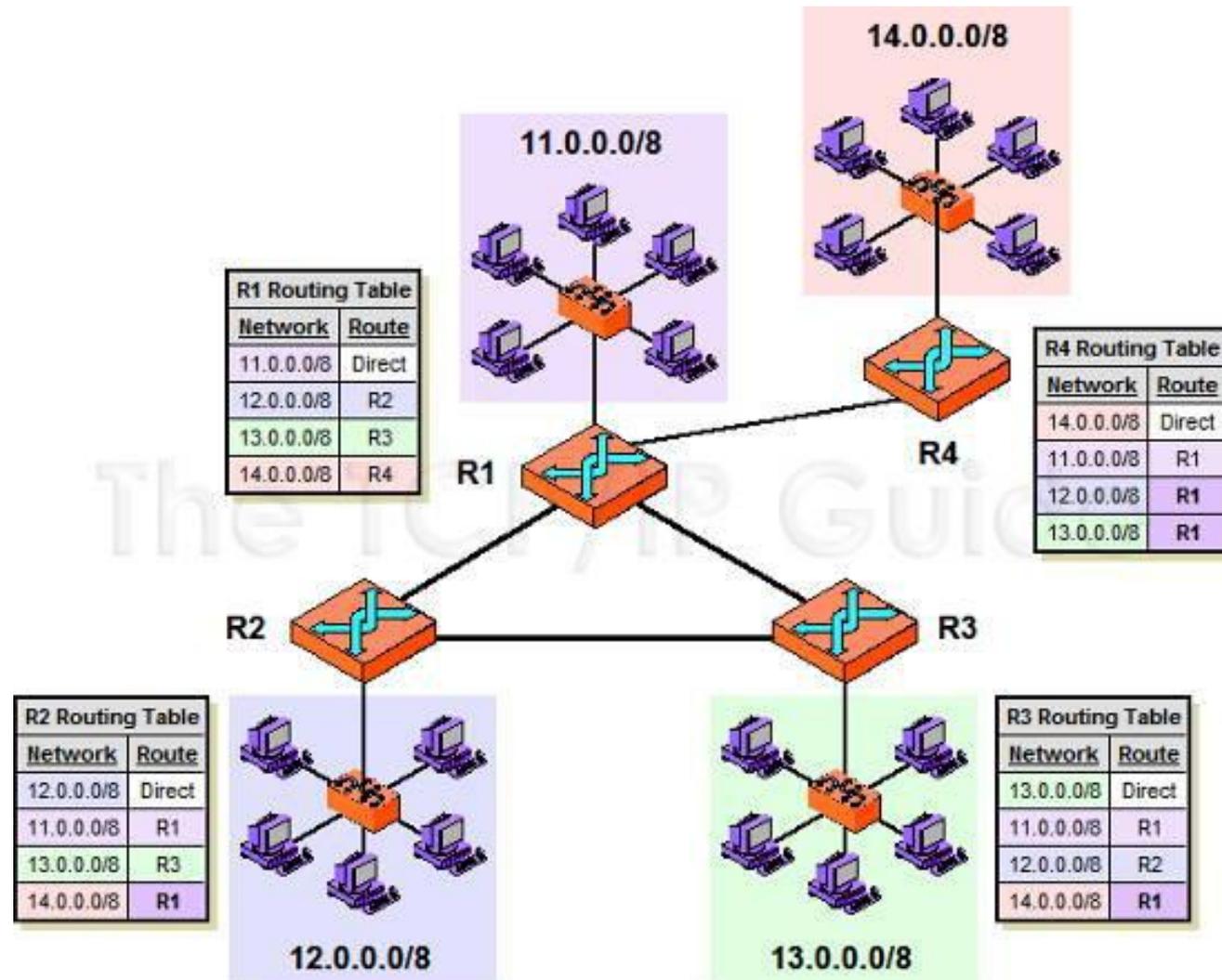
- Es:



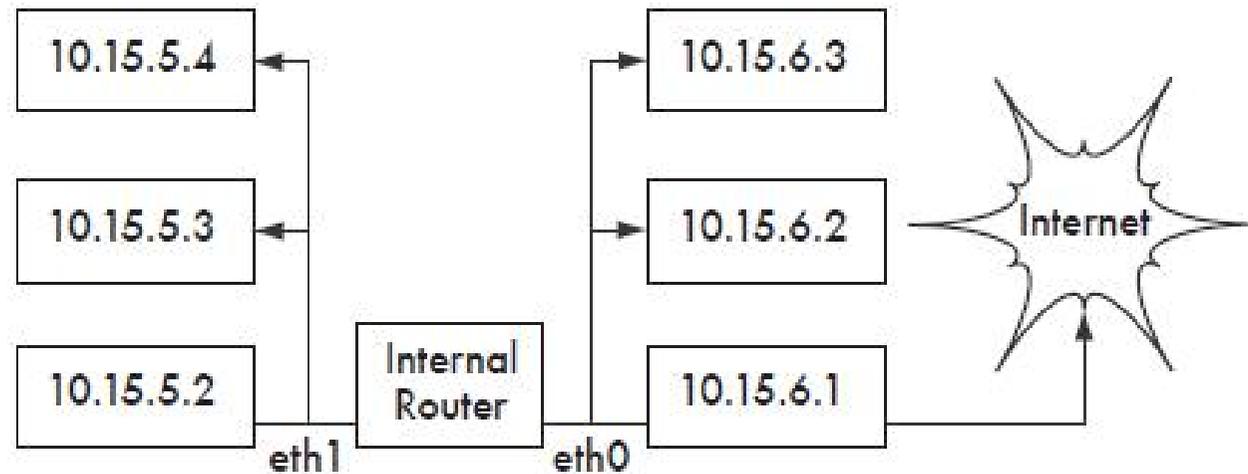
- ES:



# Un esempio più chiaro



# Gateway



Routing table for internal router:

Destination	Gateway	Genmask	Flags	Metric	Iface
10.15.5.0	*	255.255.255.0	U	0	eth1
10.15.6.0	*	255.255.255.0	U	0	eth0
default	10.15.6.1	0.0.0.0	UG	0	eth0

# Matching

- Il router esamina l'indirizzo di un host per vedere a quale sottorete appartiene.
- Sia  $X$  è l'indirizzo dell'host,  $Y$  l'indirizzo di rete e  $M$  la subnet mask relativa ad  $Y$ , se si verifica che  $X \text{ AND } M = Y \text{ AND } M$ ,  $X$  appartiene a  $Y$
- Si dice in questo caso che è il miglior matching

# Esempio di matching

Destinazione y	M	Porta uscita
195.16.1.0	255.255.255.0	2
195.13.0.0	255.255.255.0	3
10.3.0.0	255.0.0.0	4

Dalla tabella di instradamento si vuole ricavare verso quale porta sono instradati i seguenti indirizzi IP.

195.3.0.0

195.4.6.8

10.1.0.0

195.16.1.0 AND 255.255.255.0 = 195.4.6.8 AND 255.255.255.0 = 255.255.255.0 porta 2

195.13.0.0 AND 255.255.255.0 = 195.3.0.0 AND 255.255.255.0 = 255.255.0.0 porta 3

10.3.0.0 AND 255.0.0.0 = 10.1.0.0 AND 255.0.0.0 = 255.0.0.0 porta 4

# Route a costi diversi

- Può capitare che nella tabella di routing ci sia siano next hop allo stesso livello ma a costo diverso.
- L'unità di misura del costo di ogni router dipende dal protocollo utilizzato.
- Si sceglie il next hop con il costo più basso
- La politica di routing o algoritmo di routing definisce i criteri di scelta del cammino. Tramite tale politica si costruisce una tabella per l'inoltro o forwarding dei pacchetti.

# Routing statico e dinamico

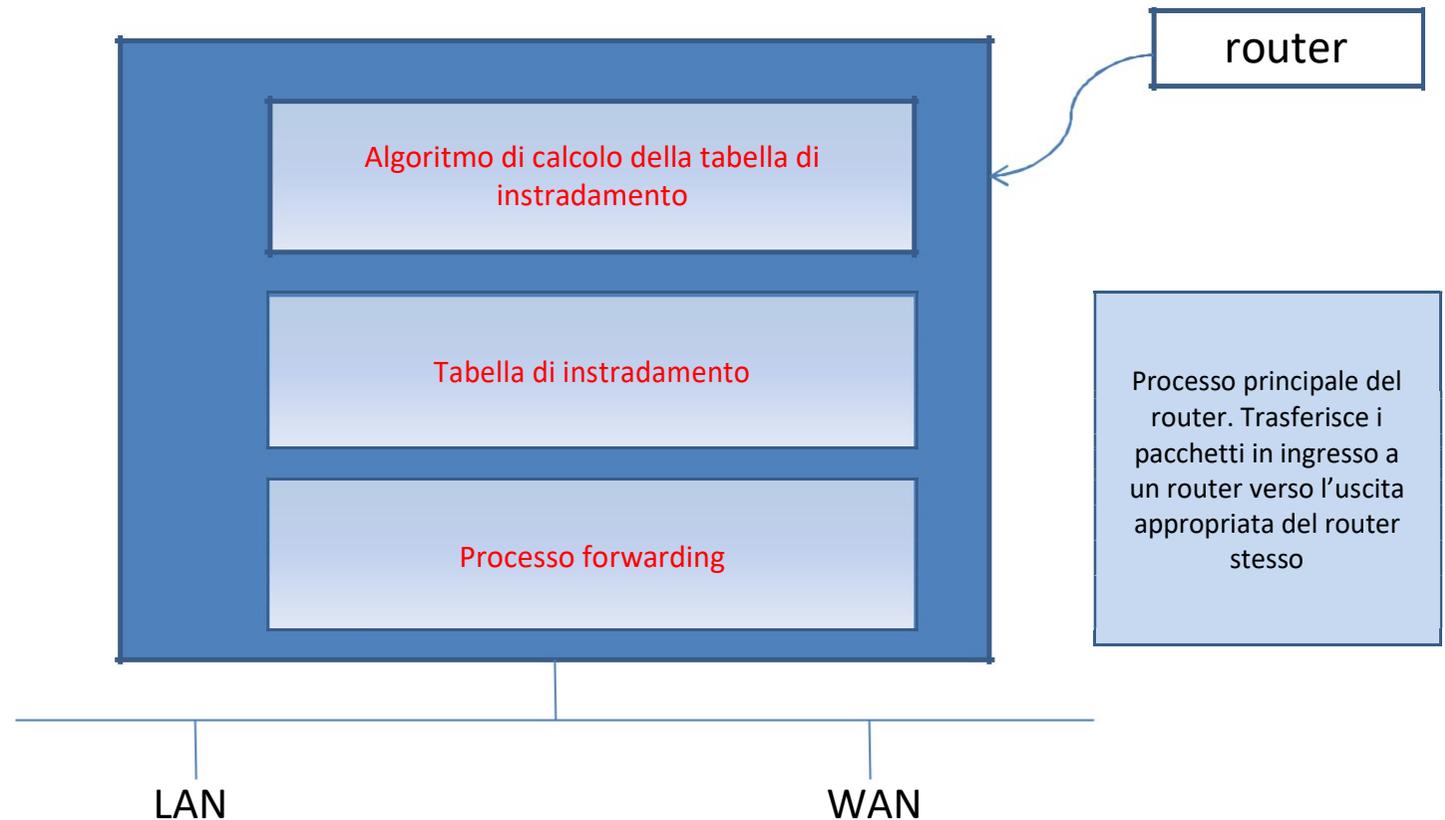
- Per il routing statico la configurazione viene fatta manualmente dall'amministratore di rete e resta tale. Le operazioni da effettuare sono le seguenti:
  - L'amministratore di linea individua manualmente la route
  - Installa la route nelle tabelle
  - I pacchetti vengono inoltrati seguendo questa route
- Nel routing dinamico le informazioni vengono ricevute dagli altri router

# Scelta dell'algoritmo di routing

- I parametri sulla quale avviene la scelta di un algoritmo di routing sono i seguenti:
  - Semplicità: i router hanno CPU e memoria finita
  - Robustezza: devono adattarsi alla topologia delle diverse reti
  - Stabilità: gli algoritmi devono produrre risultati accettabili in tempi brevi
  - Equità: deve essere garantito lo stesso trattamento e allo stesso modo per tutte le reti
  - Metrica da adottare: numero di salti, somma di costi ecc...

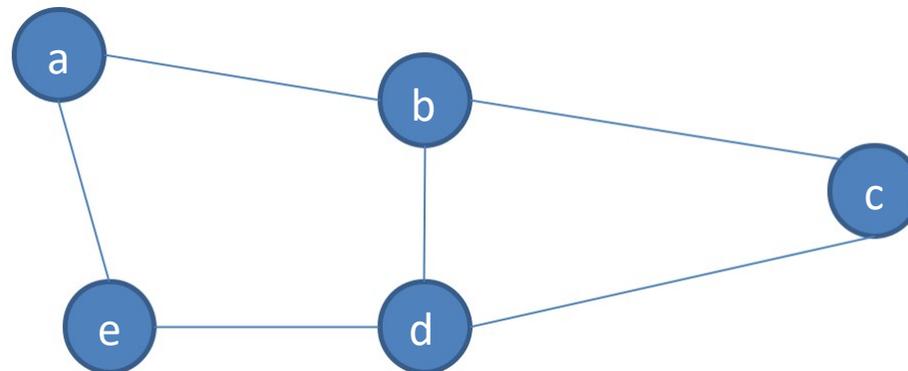
# Scelta dell'algoritmo di routing

- Nella maggior parte delle reti i due tipi di routing, statico e dinamico , operano congiuntamente.
- Le attività di un router possono essere schematizzate nel seguente modo



# Reti, grafi, alberi

- I grafi aiutano alla ricerca del cammino minimo dall'host mittente all'host destinatario
- I grafi sono fatti da nodi e archi; i nodi sono i router o le reti stesse; gli archi sono la connessione tra due router o un router e una rete



# Grafo orientato e non orientato

- Un grafo è orientato se il percorso può avvenire solo in un verso.
- Un grafo è pesato se ad ogni suo percorso, arco, viene dato un peso.
- Per cammino si intende una sequenza di vertici adiacenti
- La distanza è il cammino più breve
- Due vertici sono raggiungibili se sono collegati
- Un albero è un grafo non orientato e aciclico
- Un grafo è connesso se tutti i suoi nodi sono collegati ad un cammino

# Tabella delle adiacenze per il grafo non pesato e non orientato

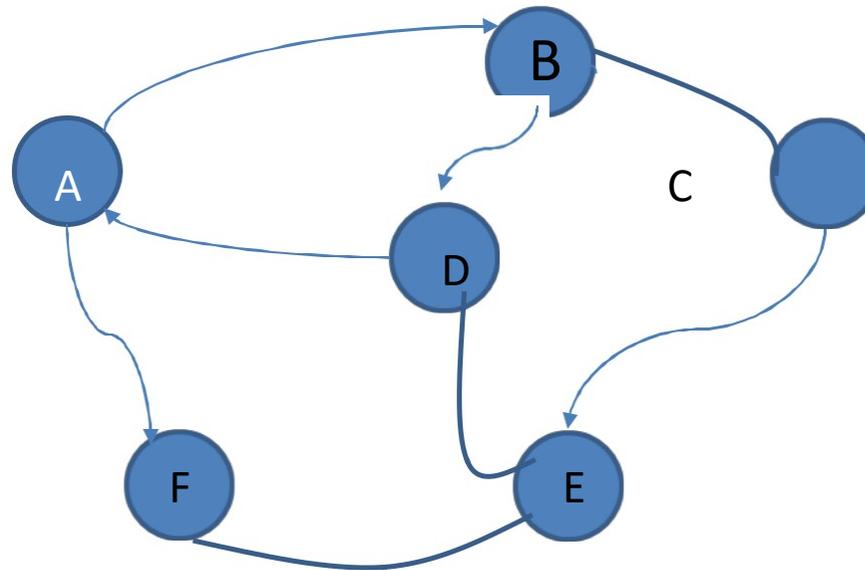
- La tabella delle adiacenze impone 1 se due router sono adiacenti, altrimenti zero
- In riferimento al grafo precedente:

	a	b	c	d	e
a	0	1	0	0	1
b	1	0	1	1	0
c	0	1	0	1	0
d	0	1	1	0	1
e	1	0	0	1	0

# Grafo orientato non pesato

- Un grafo orientato definisce il verso del percorso da un router all'altro e interdice il verso opposto

Es:



# Tabella delle adiacenze

	A	B	C	D	E	F
A	0	1	0	0	0	1
B	0	0	1	1	0	0
C	0	1	0	0	1	0
D	1	0	0	0	1	0
E	0	0	0	1	0	1
F	0	0	0	0	1	0

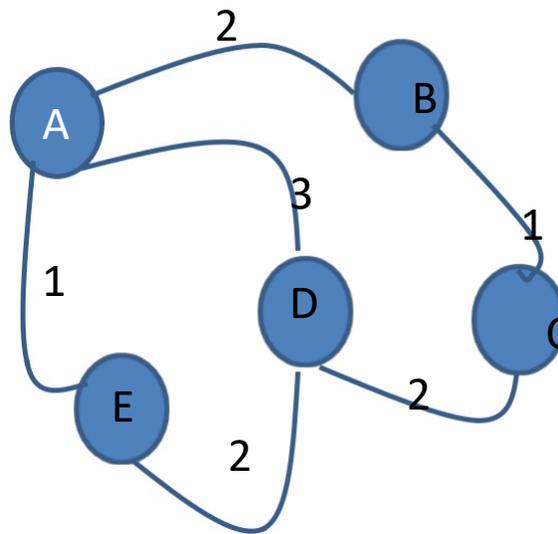
Si noti dalla tabella che il percorso da A a B è possibile ma non il viceversa. Lo stesso vale dove c'è la freccia, cioè il percorso orientato

Dove non c'è la freccia il percorso è non orientato

# Grafo pesato

- In un grafo pesato, ad ogni percorso viene dato un peso, un costo
- Per costo si intende la quantità di traffico oppure il tempo o il costo vero e proprio

Es.



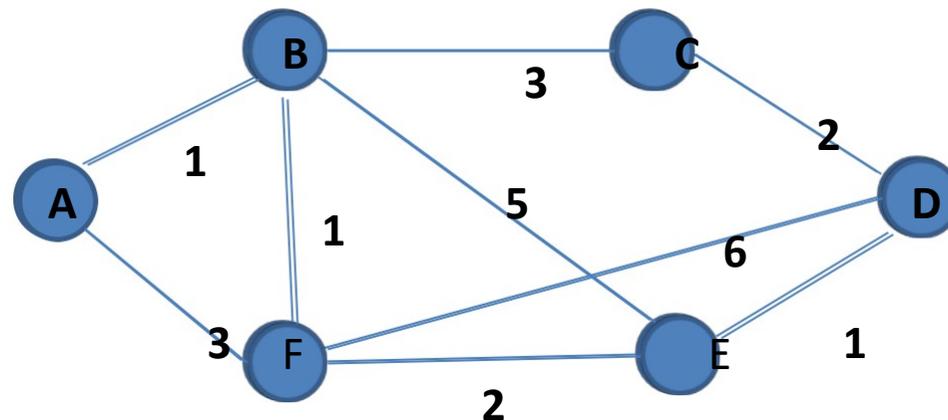
	A	B	C	D	E
A	0	2	$\infty$	3	1
B	2	0	1	$\infty$	$\infty$
C	$\infty$	1	0	2	$\infty$
D	3	$\infty$	2	0	2
E	1	$\infty$	$\infty$	2	0

# Shortest path

- Per ogni destinazione è sempre possibile scegliere il percorso minimo, quello con il peso minore
- Se il grafo non è pesato si sceglierà il percorso con meno nodi possibili
- Il percorso minimo prende il nome di albero di inoltro
- Esistono più algoritmi per individuare l'albero di inoltro. I due fondamentali e che prendono il nome dei loro ideatori:
  - Dijkstra
  - Belman -Ford

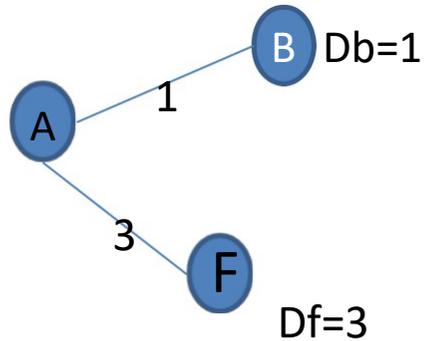
# Algoritmo statico: Dijkstra

- **Stabilito il punto di partenza e la destinazione, l'algoritmo consiste nell'assegnare di volta in volta delle etichette temporanee ai nodi adiacenti. Una volta stabilito il percorso adiacente più conveniente, si cancellano gli altri. Si parte con il seguente esempio**



# Algoritmo di Dijkstra

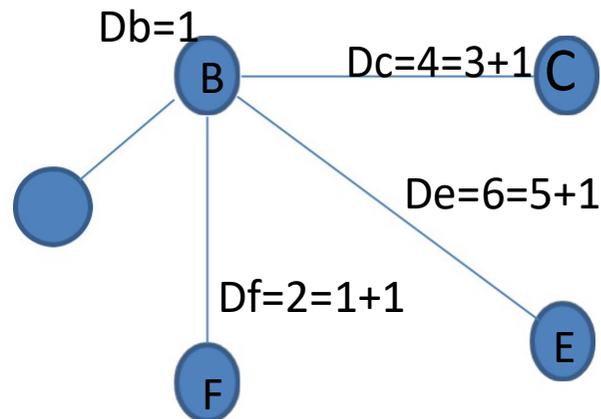
I. Scelgo di partire da A per arrivare a D



Si sceglie il percorso meno costoso, quindi AB e si elimina AF

# Algoritmo di Dijkstra

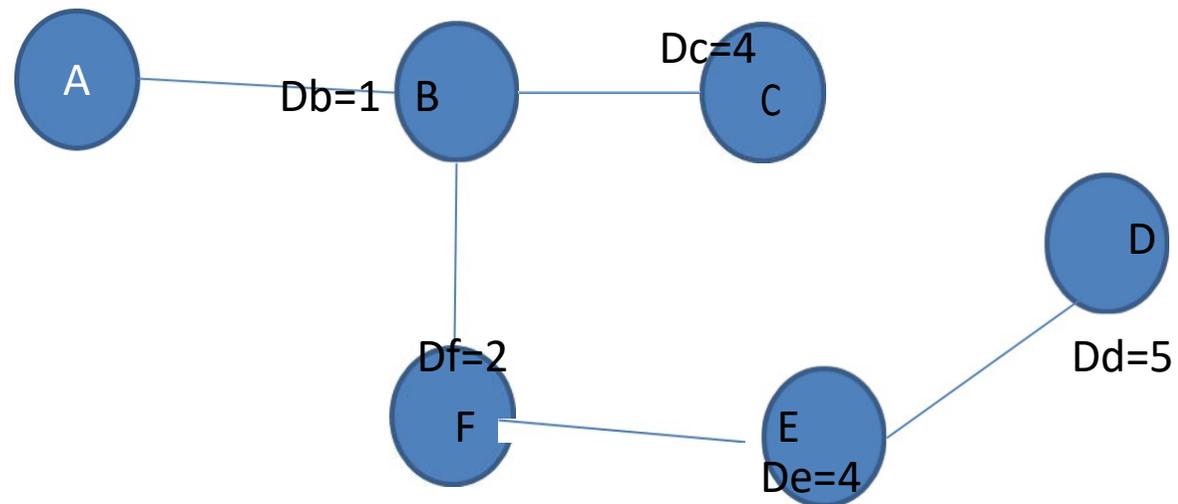
- II. Si traccia il grafo con un percorso sicuro e gli altri adiacenti da scegliere



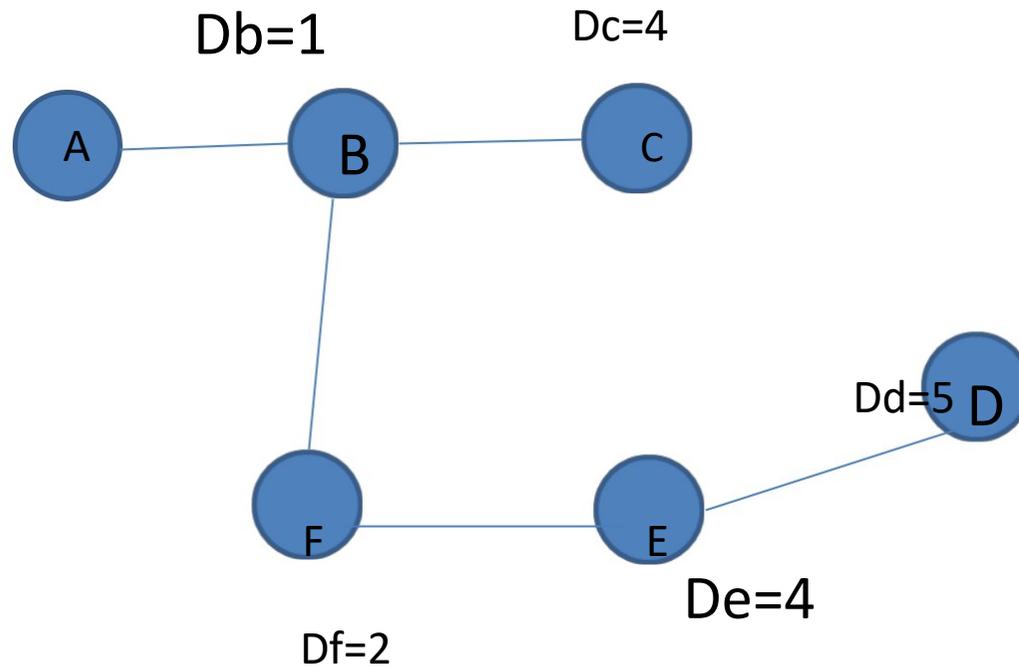
Si sceglie il percorso ABF e si cancellano gli altri

# Algoritmo di Dijkstra

III. Si procede all'ultima semplificazione



# Algoritmo di Dijkstra

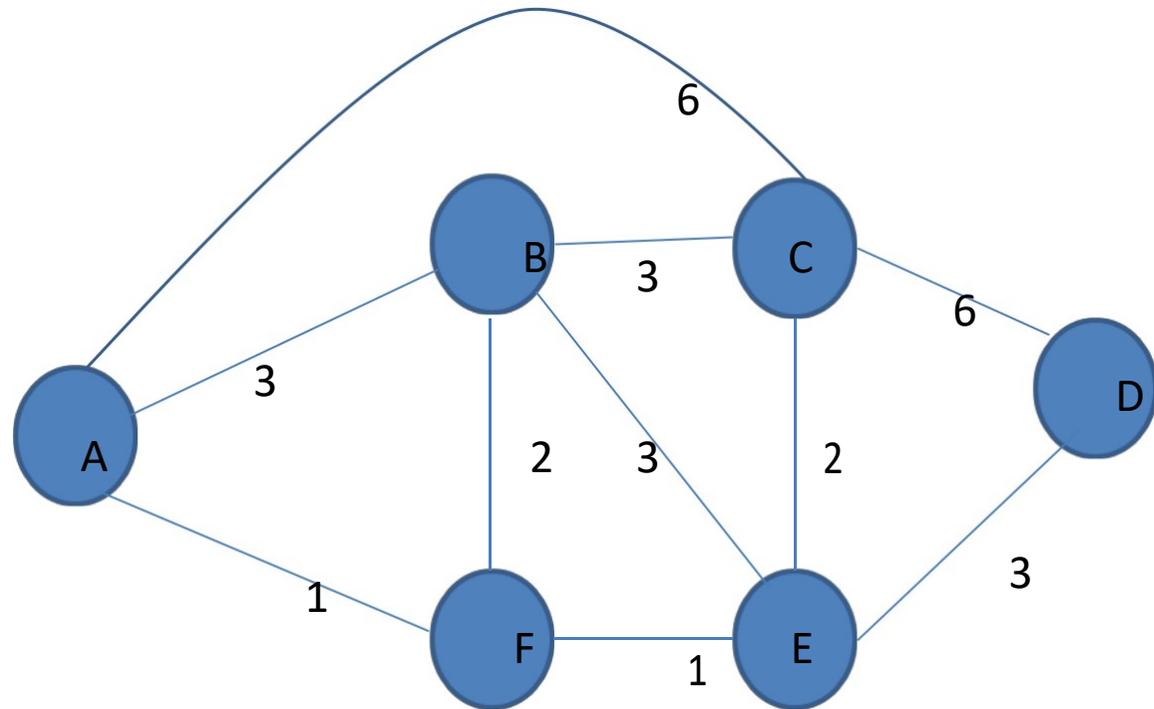


Il grafo ottenuto è un albero completo nel senso che tutti i nodi sono raggiungibili e a minor costo possibile

# Algoritmo di Bellman Ford

- L'algoritmo parte da una tabella delle adiacenze che altro non è che una tabella di next hop
- La tabella viene aggiornata grazie alle comunicazioni tra router
- Ogni router comunica a quelli adiacenti, il peso per raggiungere gli altri router adiacenti. La tabella diventa così di percorsi.

# Algoritmo di Bellman Ford



# Algoritmo di Bellmann Ford

Si scrive prima le tabelle delle adiacenze e poi quelle dei percorsi

	A	B	C	D	E	F
A	0	3	1	6	$\infty$	$\infty$
B	3	0	2	3	$\infty$	$\infty$
C	1	2	0	3	1	$\infty$
D	6	3	3	0	2	6
E	$\infty$	$\infty$	1	2	0	3
F	$\infty$	$\infty$	$\infty$	6	3	0

	A	B	C	D	E	F
A	0	3	1	6	2	5
B	3	0	2	3	5	8
C	1	2	0	3	1	4
D	6	3	3	0	2	6
E	2	5	1	2	0	3
F	5	8	4	6	3	0

# Algoritmo di Bellman Ford

- La tabella dei percorsi non è univoca perché ci sono altri percorsi possibili
- L'algoritmo di Bellman Ford sceglie tra tutti i percorsi possibili, quello di peso minimo
- Es si vuole partire da A ad F
- A vede B, C e D
- $d_{AB}=3$ ,  $d_{AC}=1$ ,  $d_{AD}=1$
- $d_{AF} = \min(d_{AB} + d_{BF} ; d_{AC} + d_{CF} ; d_{AD} + d_{DF} )$