

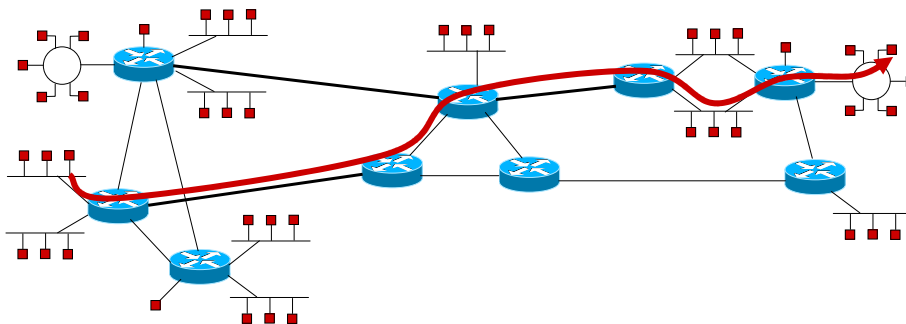


# Il routing nelle reti IP

A.A. 2005/2006

Walter Cerroni

## IP: instradamento dei datagrammi



- **Routing** : scelta del percorso su cui inviare i dati
  - i **router** formano una struttura interconnessa e cooperante: i datagrammi passano dall'uno all'altro finché raggiungono quello che può consegnarli direttamente al destinatario

2

## Tabelle di routing

- La funzione di instradamento viene eseguita dai router tramite l'utilizzo di apposite **tabelle di routing**
  - sono presenti in ogni nodo della rete
  - per ogni indirizzo di destinazione del tipo Net-ID/Netmask indicano:
    - l'interfaccia da usare per raggiungere quella destinazione
    - la modalità di consegna (diretta o indiretta)
    - il gateway (**next-hop**) da usare nel caso di consegna indiretta
    - il costo (o metrica) per raggiungere quella destinazione
  - possono essere presenti direzioni alternative corrispondenti a costi diversi
- I datagrammi IP vengono instradati indipendentemente l'uno dall'altro in modalità **connection-less**
  - la ricerca nelle tabelle (**table look-up**) avviene sulla base dell'indirizzo IP di destinazione del pacchetto da instradare

3

## Esempio di tabella di instradamento

Router con 3 interfacce:

eth0 = 137.204.57.253/24

ppp0 = 10.0.0.9/30

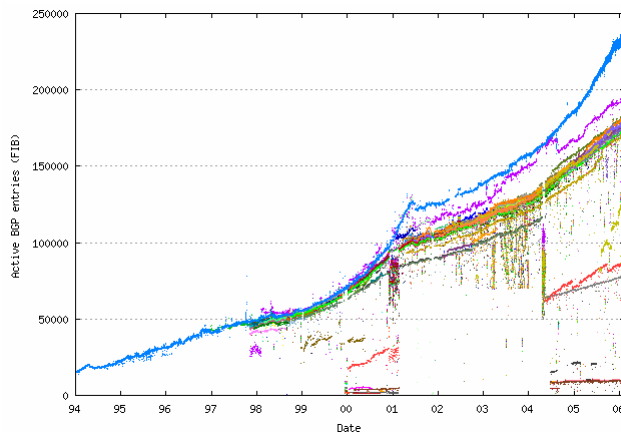
ppp1 = 10.0.0.13/30

Destinazione	Netmask	Gateway	Metrica	Interfaccia
0.0.0.0	0.0.0.0	137.204.57.254	1	eth0
10.0.0.8	255.255.255.252	0.0.0.0	1	ppp0
10.0.0.12	255.255.255.252	0.0.0.0	1	ppp1
137.204.56.0	255.255.255.0	10.0.0.14	1	ppp1
137.204.57.0	255.255.255.0	0.0.0.0	1	eth0
137.204.58.0	255.255.254.0	10.0.0.10	1	ppp0
137.204.59.18	255.255.255.255	0.0.0.0	1	eth0
137.204.121.128	255.255.255.128	10.0.0.14	1	ppp1
137.204.121.128	255.255.255.128	10.0.0.10	2	ppp0
192.168.8.0	255.255.252.0	10.0.0.14	1	ppp1

4

## Dimensioni reali delle tabelle di routing

- Crescita del numero di voci attive nelle tabelle BGP dal '94 ad oggi



Fonte: <http://bgp.potaroo.net/>

<http://bgp.potaroo.net/as1221/bgptable.txt> → 51 MB di file txt

5

## Algoritmi di table lookup

- La consultazione della tabella di routing avviene per ogni pacchetto secondo la tecnica **longest-prefix match**
  - in presenza di più scelte possibili, si sceglie la riga che ha il maggior numero di bit in comune con l'indirizzo di destinazione (priorità alle route più specifiche)
    - 137.204.0.0/16
    - 137.204.57.0/24
    - 137.204.57.128/25
- Ad alte bit-rate è necessario un elevato numero di look-up/sec
  - la velocità di look-up dipende dalle dimensioni della tabella e dal numero di accessi alla memoria
  - le voci contenute nella tabella subiscono modifiche su scale temporali relativamente lunghe (dell'ordine dei minuti)
- Si adottano specifiche strutture dati ad albero (**Patricia Trie**)
  - look-up veloci con pochi accessi alla memoria
  - modifiche più costose in termini di tempi di elaborazione
  - strutture compatte da inserire in memorie veloci (processor cache)

6

## Il trasferimento dei datagrammi IP

---

In ciascun nodo ha due componenti funzionali distinte

### Instradamento (routing)

- Creazione di tabelle di instradamento tramite:
  - scambio di informazioni tra router (protocolli di routing)
  - elaborazione locale (algoritmi di routing)

### Inoltro (forwarding)

- Algoritmo di forwarding
  - lettura dell'intestazione IP ed estrazione dell'indirizzo di destinazione
  - consultazione della tabella di instradamento (**table look-up**)
  - identificazione dell'interfaccia di uscita su cui inviare il datagramma
- Switching
  - trasferimento fisico dei datagrammi da ingresso a uscita

7



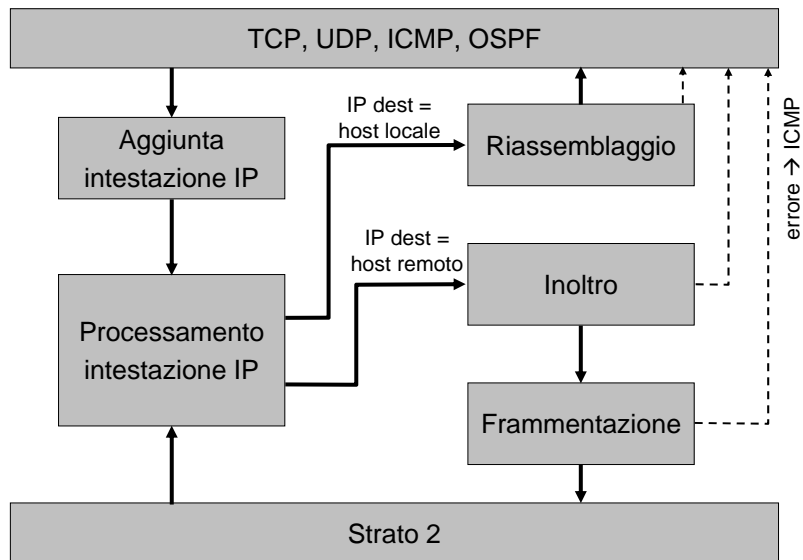
## Architetture di router IP

---

A.A. 2005/2006

Walter Ceroni

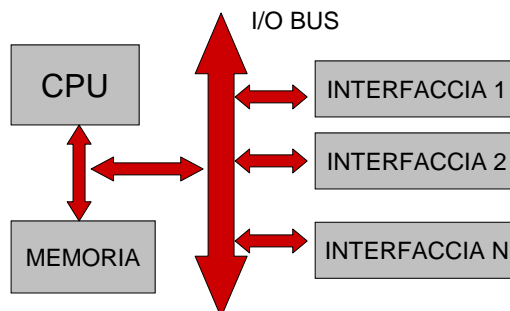
## Schema funzionale di un router



9

## Architetture di router di prima generazione

- Workstation con schede di rete, bus e CPU centrale
  - i pacchetti ricevuti dalle interfacce vengono trasferiti nella memoria (DMA) tramite il bus
  - la CPU elabora le intestazioni e consulta la tabella di routing, inoltre esegue i protocolli di routing
  - i pacchetti elaborati vengono trasferiti sulle interfacce ancora tramite il bus



10

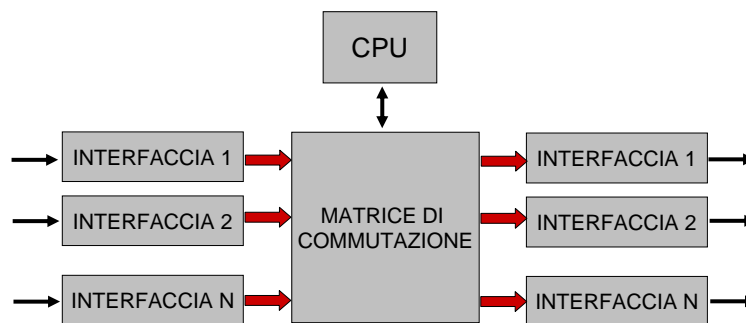
## Prestazioni

- Sono limitate dal fatto che tutti i pacchetti devono transitare per un solo punto di contesa rappresentato tipicamente dal bus di I/O, che viene attraversato due volte per ciascun trasferimento
  - un processore con un bus di I/O a 1Gbit/s può supportare fino a 5 link a 100 Mbit/s o 3 link a 155 Mbit/s, ma neanche un link a 622 Mbit/s
- Inoltre, in presenza di tabelle di routing di grandi dimensioni, il tempo di accesso alla memoria può risultare un collo di bottiglia
- Occorrono quindi soluzioni più veloci, tipicamente realizzate con hardware specializzato (ASIC)

11

## Architetture attuali di router

- La **CPU** calcola la tabella di routing, esegue i protocolli e gestisce il sistema nel suo complesso
- Le **interfacce d'ingresso** memorizzano i pacchetti ed eseguono il look-up su copie locali della tabella (più o meno complete)
- La **matrice di commutazione** permette il trasferimento diretto e contemporaneo di più pacchetti
- Elaborazioni realizzate in hardware (ASIC)



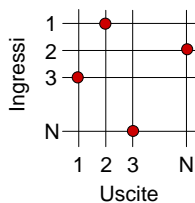
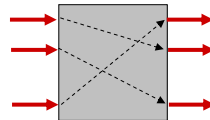
12

## Matrici di commutazione $N \times N$

- Permettono più trasferimenti in parallelo
- Esempi:

### – crossbar

- $N^2$  punti di incrocio aperti/chiusi da un dispositivo di controllo (**scheduler**)
- matrice monostadio
- non bloccante: è sempre possibile instaurare un percorso tra due terminazioni libere qualunque sia lo stato delle connessioni in atto
- velocità di commutazione limitata dalla velocità dello scheduler



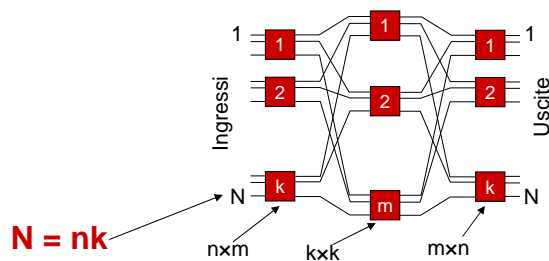
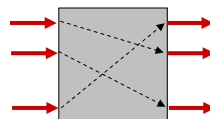
13

## Matrici di commutazione $N \times N$

- Permettono più trasferimenti in parallelo
- Esempi:

### – rete di Clos

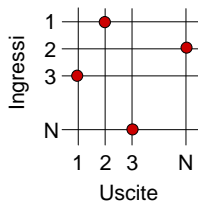
- matrice multistadio costituita da elementi di commutazione non bloccanti di dimensioni inferiori
- riduce il numero di punti di incrocio necessari (complessità)
- non bloccante se si rispettano alcune condizioni



14

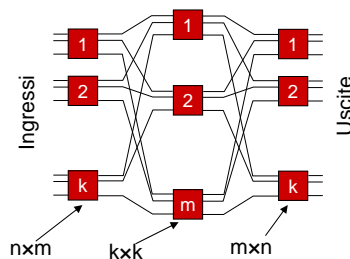
## Matrice crossbar e rete di Clos: complessità

Crossbar



complessità =  $N^2$

Rete di Clos simmetrica a 3 stadi



- $N = nk$
- non bloccante se  $m \geq 2n - 1$
- complessità =  $2knm + mk^2$
- con una opportuna scelta di  $n$  e  $m$   
 $\rightarrow$  complessità =  $O(N^{3/2})$

15

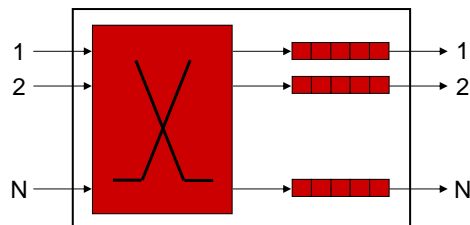
## Posizionamento delle memorie

- Nelle reti a commutazione di pacchetto in ogni nodo tipicamente si memorizzano i pacchetti prima di trasmetterli in uscita (**store and forward**)
  - memorie nelle interfacce di ingresso durante l'elaborazione dell'intestazione e il look-up
  - memorie nelle interfacce di uscita prima della trasmissione
- L'utilizzo di matrici di commutazione (anche se non bloccanti) per trasferire pacchetti necessita di code (**buffer**) per risolvere le contese tra pacchetti diretti verso la stessa porta di uscita
  - alternative:
    - **accodamento in uscita** (Output Queuing – OQ)
    - **accodamento in ingresso** (Input Queuing – IQ)
    - **accodamento virtuale in uscita** (Virtual Output Queuing – VOQ)
    - **accodamento a memoria comune** (Shared Buffer – SB)
      - varie combinazioni dei precedenti
  - velocità di commutazione limitata dalla velocità di accesso alla memoria che implementa le code

16



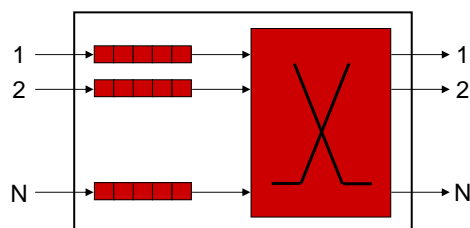
## Accodamento in uscita



- I pacchetti sono accodati **dopo** essere stati commutati
- I pacchetti che trovano una linea di uscita occupata vengono accodati in attesa che essa si liberi
- Se arrivano N pacchetti destinati alla stessa uscita, la matrice di commutazione deve inserirli tutti nella relativa coda
- La velocità di commutazione della matrice deve essere almeno pari alla somma delle velocità di trasmissione dei pacchetti delle N linee di ingresso (**speed-up** con fattore N)
  - questa soluzione può risultare molto costosa con linee ad elevata bit-rate

17

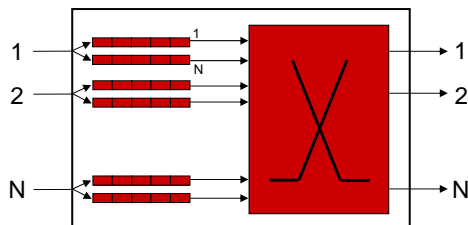
## Accodamento in ingresso



- I pacchetti sono accodati **prima** di essere commutati
- Il pacchetto in testa alla coda viene trasferito quando l'uscita a cui è destinato risulta libera
- La matrice di commutazione non deve introdurre nessuno speed-up
- Problema di **blocco Head-Of-Line** (HOL)
  - il pacchetto di testa attende che si liberi l'uscita a cui è destinato
  - tutti gli altri pacchetti dietro di lui devono attendere che questo venga servito, anche se la loro uscita è disponibile (code **FIFO**)
  - throughput massimo = 58%

18

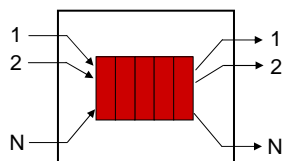
## Accodamento virtuale in uscita



- Per ogni ingresso c'è una coda dedicata ad ogni uscita
- I pacchetti sono accodati **prima** di essere commutati ma **dopo** aver selezionato la coda di uscita (coda di uscita virtuale)
- Si elimina il problema del blocco HOL
  - throughput massimo = 100%
- La matrice di commutazione non deve introdurre nessuno speed-up
- Sono necessarie  $N^2$  code
- E' necessario uno scheduler più complesso

19

## Accodamento a memoria comune



- I pacchetti sono inseriti tutti in un'**unica memoria comune**
- All'interno della memoria si ha una organizzazione in code logiche, ciascuna associata a una uscita
- Poiché più code logiche vengono combinate in un'unica coda fisica, si ha una migliore utilizzazione della memoria
- Si perde un pacchetto solo quando la somma delle occupazioni delle code logiche eguaglia la memoria complessiva
- Si complica la gestione della memoria
  - deve essere ad accesso casuale (non bastano shift-register)
  - richiede l'inserimento/estrazione di N pacchetti alla volta

20

## Tipologie commerciali di router

---

- **Access router**
  - utilizzati dai provider (ISP) per fornire accesso ad utenti domestici e a piccole imprese
    - elevato numero di porte a bit-rate medio-bassa (50 kbps ÷ 10 Mbps)
    - grande varietà di protocolli e tecnologie di accesso (PPP, ADSL, FTTH, ...)
- **Enterprise/campus router**
  - utilizzati per interconnettere le infrastrutture di rete di imprese medio-grandi o di grossi enti (università)
    - limitato numero di porte ad alta bit-rate (10 Mbps ÷ 1 Gbps)
    - costo non troppo elevato
- **Backbone router**
  - utilizzati per interconnettere reti su scala regionale/globale
    - poche porte ad elevatissima bit-rate (1 Gbps ÷ 10 Gbps)
    - costo elevato ma alta affidabilità ed efficienza