



Architettura degli switch

Mario Baldi

Politecnico di Torino

<http://staff.polito.it/mario.baldi>

Pietro Nicoletti

Studio Reti

<http://www.studioreti.it>

Basato sul capitolo 8 di:

M. Baldi, P. Nicoletti, "Switched LAN", McGraw-Hill, 2002, ISBN 88-386-3426-2

Nota di Copyright

Questo insieme di trasparenze (detto nel seguito slide) è protetto dalle leggi sul copyright e dalle disposizioni dei trattati internazionali. Il titolo ed i copyright relativi alle slide (ivi inclusi, ma non limitatamente, ogni immagine, fotografia, animazione, video, audio, musica e testo) sono di proprietà degli autori indicati a pag. 1.

Le slide possono essere riprodotte ed utilizzate liberamente dagli istituti di ricerca, scolastici ed universitari afferenti al Ministero della Pubblica Istruzione e al Ministero dell'Università e Ricerca Scientifica e Tecnologica, per scopi istituzionali, non a fine di lucro. In tal caso non è richiesta alcuna autorizzazione.

Ogni altra utilizzazione o riproduzione (ivi incluse, ma non limitatamente, le riproduzioni su supporti magnetici, su reti di calcolatori e stampate) in toto o in parte è vietata, se non esplicitamente autorizzata per iscritto, a priori, da parte degli autori.

L'informazione contenuta in queste slide 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. Gli autori non assumono alcuna responsabilità per il contenuto di queste slide (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 slide.

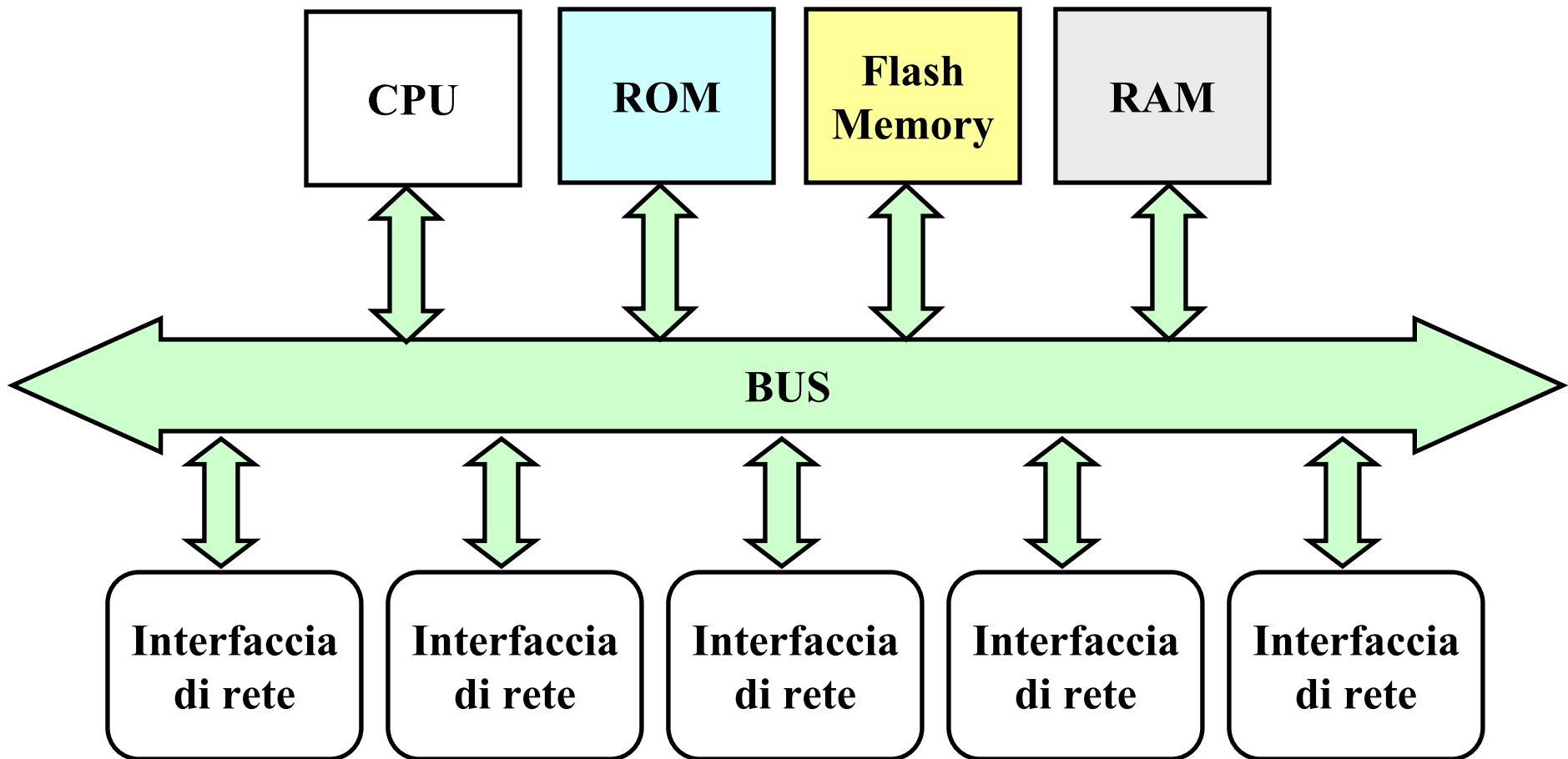
In ogni caso questa nota di copyright non deve mai essere rimossa e deve essere riportata anche in utilizzi parziali.

Bridge o Switch?

- I due termini sono spesso utilizzati in modo intercambiabile
 - Funzionalmente identici
- *Bridge*: apparato di internetworking di livello 2
 - Inoltra trame MAC tra LAN separate
- *Switch*: termine commerciale introdotto per enfatizzare la velocità dell'apparato
 - Inoltra normalmente realizzato da hardware (ASIC)
 - Maggiore numero di porte
 - Più elevato throughput aggregato
 - Stesse funzionalità

Tradizionale architettura del bridge

Bassa scalabilità → basso numero di porte



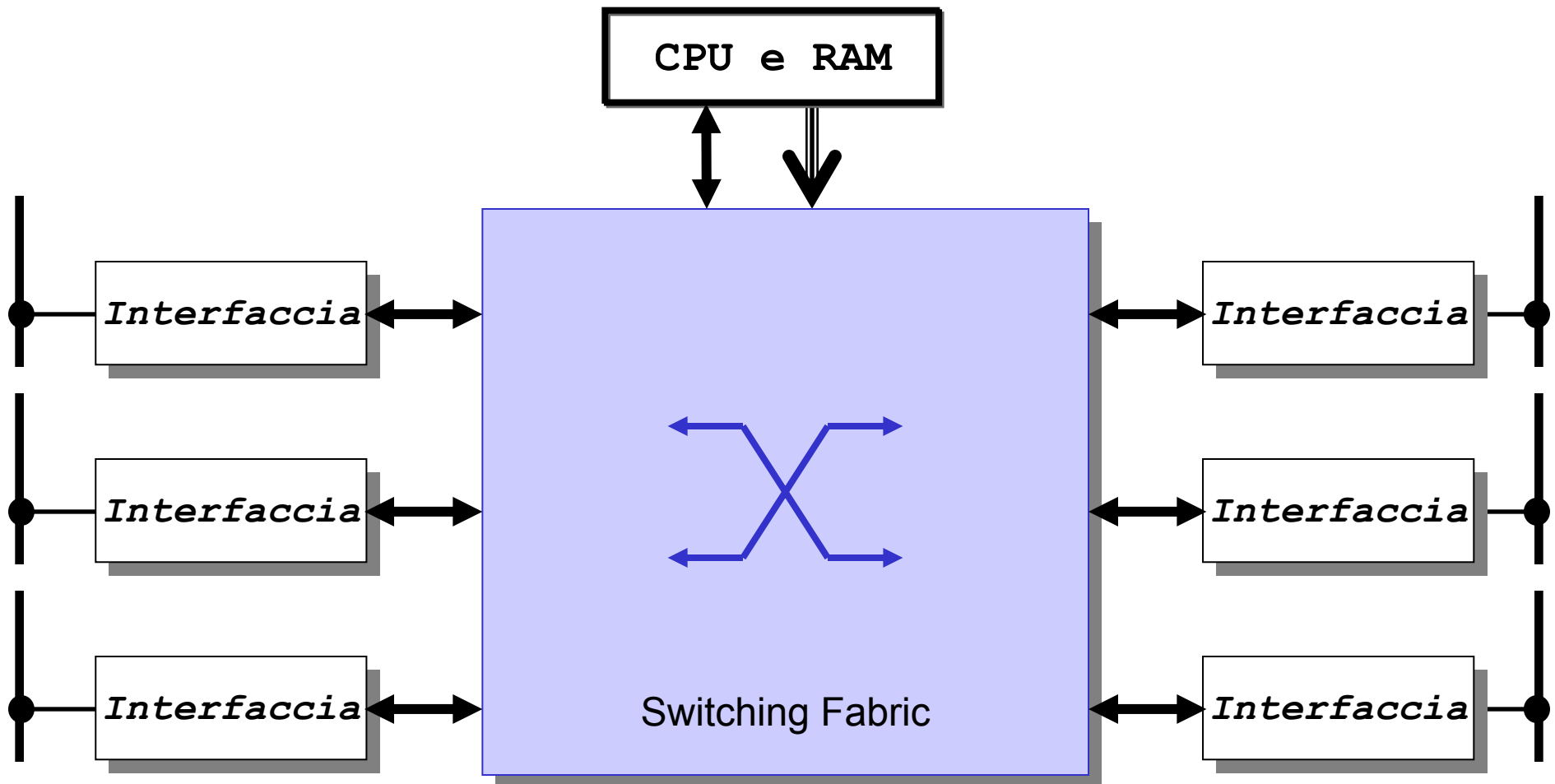
Limiti di scalabilità

- Un elevato numero di porte è indispensabile nella realizzazione di switched LAN
 - Al limite una porta per ogni stazione della rete
- Scalabilità del bridge è limitata da colli di bottiglia
 - Processore
 - Memoria
 - Bus
- Aumentare il numero di porte (o la loro velocità) di un fattore N richiede lo stesso miglioramento in
 - Capacità elaborativa del processore
 - Velocità di accesso alla memoria contenente informazioni di routing
 - Capacità di trasferimento del bus

Come superare questi limiti

- Distribuire funzionalità tradizionalmente centralizzate
 - Elaborazione
 - Presenza di vari processori
 - Commutazione
 - Matrice di commutazione (switching fabric)
 - Svariati cammini contemporanei tra ingressi e uscite
 - Space switching invece/oltre che time switching (come nel bus)
- Utilizzo di hardware specializzato
 - Progettazione ad hoc
 - Application Specific Integrated Circuit (ASIC)
 - Meno flessibile, ma ottimizzato (più veloce)

Architettura degli switch



Distribuzione delle funzionalità

- Processore centrale: controllo
 - Esecuzione dello spanning tree protocol
 - Riconfigurazione della switching fabric
 - Management
- Processori di interfaccia
 - Inoltro pacchetti
 - Parsing pacchetto
 - Decisione di routing
 - Eventuale modifica del pacchetto
- Scalabilità: ogni processore di interfaccia elabora solo i pacchetti ricevuti da quella interfaccia
 - Aumentando il numero di interfacce si aumenta anche il numero di processori

Problematiche

- Aggiornamento e distribuzione di informazioni
 - Filtering database deve essere acceduto dai processori di interfaccia
 - Complesse tecniche di condivisione e sincronizzazione
 - Database centralizzato
 - Copie locali (cache)
- Coordinamento tra i processori di interfaccia e il processore centrale
 - Politiche di controllo della switching fabric
- Tecniche sofisticate (e proprietarie) sviluppate dai costruttori negli anni
 - Opportunità di differenziazione

Switching fabric

- Bus
- Crossbar
- Rete multistadio

Switching fabric *non-bloccante*

È in grado di trasferire le trame ricevute su ogni interfaccia di ingresso sulla rispettiva interfaccia di uscita purché non già occupata in un trasferimento

Bus

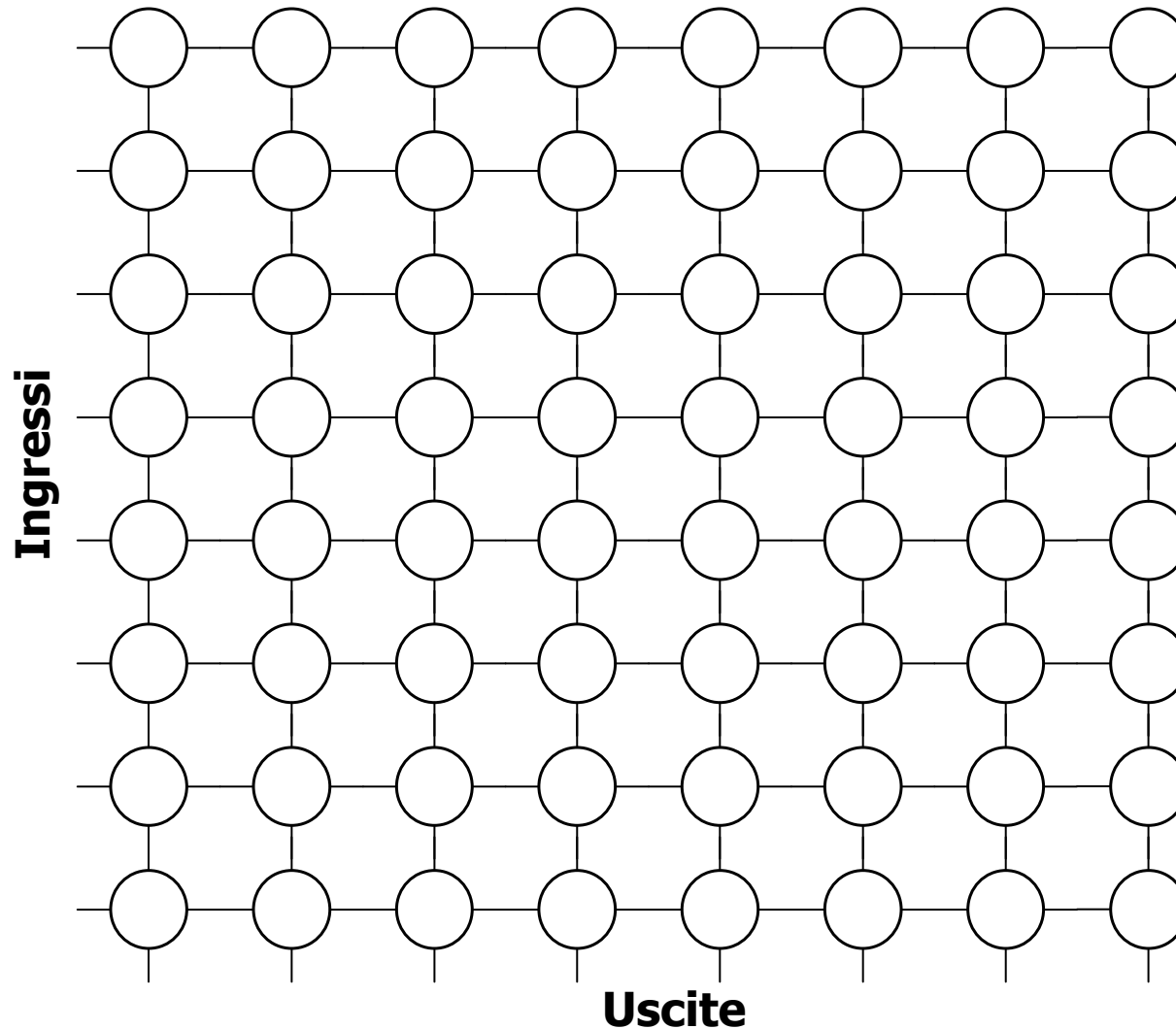
- Intrinsecamente bloccante → *speedup*
 - Capacità uguale alla capacità aggregata delle interfacce
 - Esempio: 64 interfacce ad 1 Gb/s → bus a 64 Gb/s
- Scalabilità limitata: aumentando le interfacce
 - Si deve aumentare la capacità del bus
 - Aumenta la lunghezza del bus
 - Maggiore sensibilità alle interferenze elettromagnetiche
- Soluzione: Aumentare parallelismo
 - La velocità di trasmissione su ogni linea resta limitata
 - Aumenta la complessità dei connettori
 - Problemi di interferenza tra le linee
 - Aumenta la granularità dei trasferimenti
 - Possibili inefficienze

Crossbar

- Switching fabric non bloccante per eccellenza
 - In ogni istante può collegare qualsiasi ingresso a qualsiasi uscita non occupata
- Distribuisce i pacchetti da trasferire su percorsi diversi
 - Space switching
 - Speedup non indispensabile
 - Capacità di trasferimento da ingresso ad uscita può pari alla capacità di ogni interfaccia
 - Capacità di trasferimento aggregata pari alla capacità aggregata delle interfacce
- Visione logica come rete di commutatori elementari



Crossbar: visione concettuale

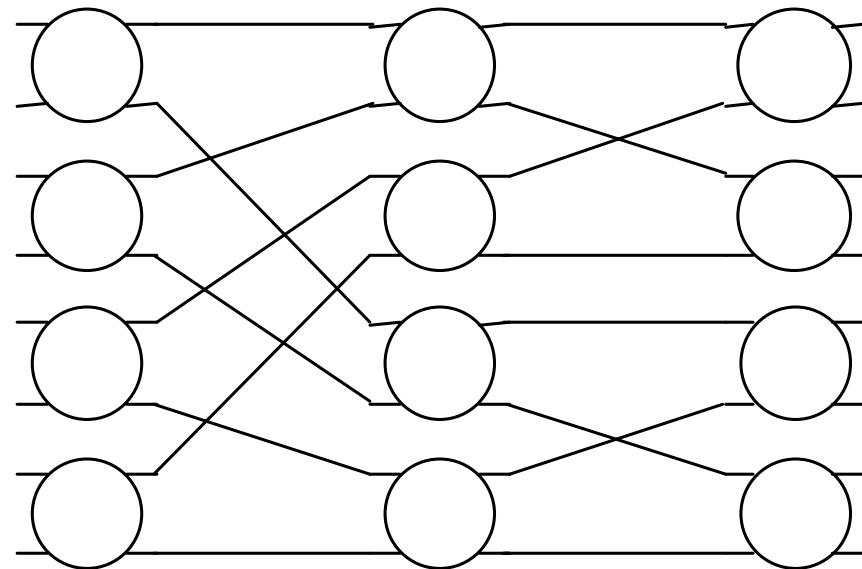


Complessità
quadratica:

10 volte più interfacce →
100 volte più commutatori

Reti multistadio

- Migliore scalabilità rispetto a crossbar
- Clos
 - Non bloccante
- Banyan
 - Minimo numero di elementi: $o(N \log N)$
 - Massima scalabilità
 - Bloccante



Switching fabric non bloccante: tutto qui?

- Trame ricevute da ingressi distinti non possono essere trasferite contemporaneamente alla stessa uscita
- Una è trasferita, le altre sono memorizzate agli ingressi

Q: Come evitare la memorizzazione in ingresso?

A: Aumentando la velocità di trasferimento:

- Si può spostare una trama da ogni ingresso ad una uscita nel tempo di ricezione di una trama
- Capacità aggregata di trasferimento pari alla capacità aggregata delle interfacce

Scalabilità?

Speedup

- Capacità di trasferimento più alta della capacità delle interfacce

- Nel caso peggiore

Switching fabric non bloccante
 +
 Speedup pari al numero di ingressi
 =
 no memorizzazione in ingresso

- In teoria

Switching fabric non bloccante
 +
 Speedup pari a 2
 =
 no congestione in ingresso

- Assunzioni sulla distribuzione di traffico: realistiche?
 - Complessi algoritmi di gestione delle code in ingresso (input queuing)
- Speedup ha impatto sulla circuiteria di interfacciamento
 - Per esempio, memoria sulla scheda di uscita

La giusta (?) via sta nel mezzo

- Alto speedup
 - Memorizzazione all'uscita (output buffering) → minore complessità
 - Minore scalabilità della switching fabric
- Basso speedup
 - Memorizzazione all'ingresso (input buffering) → maggiore complessità
 - Complessi algoritmi per la gestione delle code (scheduling)
 - Maggiore scalabilità della switching fabric

Soluzione di compromesso

- Speedup limitato - spesso inferiore a 2
- Buffer in ingresso e in uscita (combined I/O buffering)
- Gestione delle code non ottimale (ma implementabile)

Tutti seguono questa via?

Obiettivi

- Minimizzare la complessità
- Massimizzare la scalabilità
- Offrire prestazioni in genere accettabili

Soluzione

- Basso speedup (eventualmente 1)
- Code solo in uscita o scheduling elementare in ingresso
- Switching fabric eventualmente bloccante

Risultato

- Prestazioni soddisfacenti con profili di traffico reali
 - Bassa probabilità di contesa per la stessa uscita
 - Basso carico medio sulle interfacce

Switching fabric non bloccante + speedup: tutto qui?

No, se si vuole garantire la qualità del servizio!

- Eliminare la contesa per l'interfaccia di uscita non elimina la contesa per la trasmissione
 - Non si può trasmettere più di una trama alla volta
 - Una trama è trasmessa, le altre sono memorizzate
- Il servizio risultante
 - Dipende dal numero di trame in contesa
 - Dipende dal profilo *istantaneo* di traffico
- Aumentare la velocità delle interfacce non risolve il problema in generale
 - Aumenta anche la velocità di ricezione!!!!

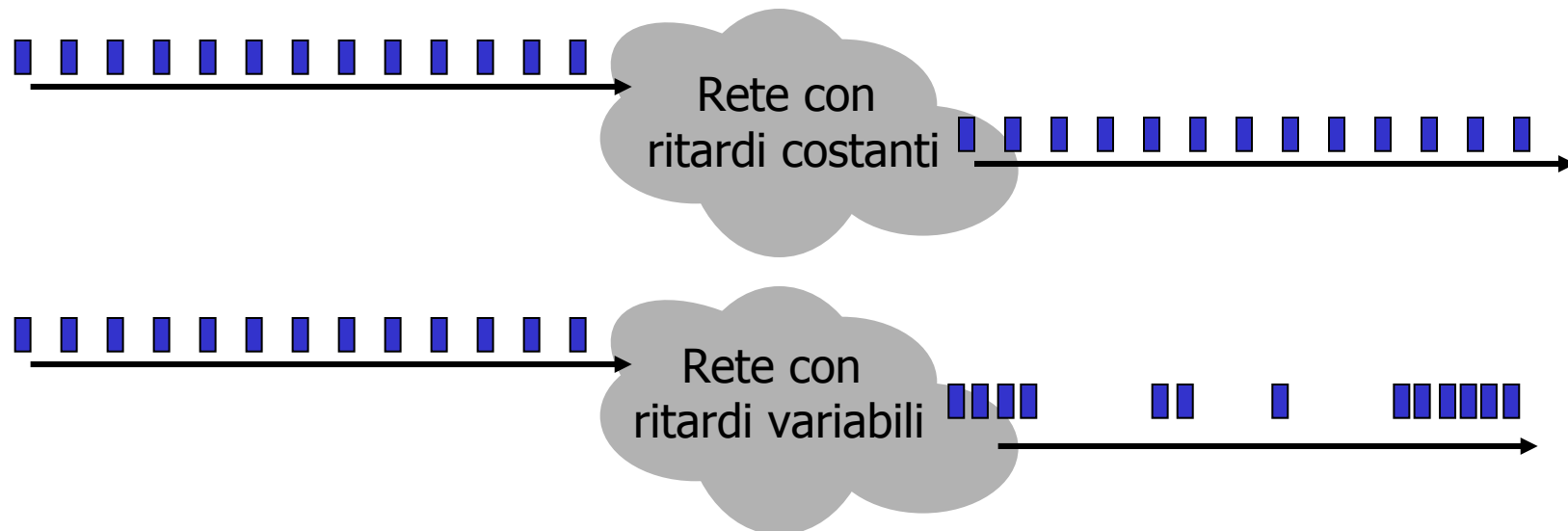
Le conseguenze e i loro rimedi

- Scarto di trame
 - Buffer sufficientemente grandi alleviano il problema
- Ritardi variabili
 - Accodamento differenziato e algoritmi di scheduling
 - Scegliere nel buffer il prossimo pacchetto da trasmettere in modo ottimale (?)
 - Algoritmi più sofisticati offrono migliore controllo sul ritardo
 - Normalmente non si vogliono switch di livello 2 complicati
 - Limitazione sulla quantità di trame in contesa (admission control)
 - Normalmente non utilizzato negli switch di livello 2

Le applicazioni real-time

Tempistiche di ricezione influenzano il funzionamento

- Voce, telefonia, musica, video, videoconferenza
- Sempre più utilizzate sulle reti locali e non
- Segnale originale è campionato ad intervalli regolari
- Per avere buona qualità i campioni devono essere riprodotti con la stessa regolarità



Controllo dei ritardi

- **Replay buffer**
 - **Alla destinazione**
 - Non richiede modifiche agli apparati di rete
 - Può essere implementato nell'applicazione stessa
 - **Aumenta i ritardi: non adatto per applicazioni interattive**
- **Gestione avanzata delle code**
 - **Soluzione alla radice del problema**
 - **Code differenziate**
 - **Algoritmi di scheduling sofisticati**
 - **Controllo del traffico**
 - Network engineering
 - Traffic engineering
 - Prenotazione delle risorse (admission control)

IEEE 802.1p