

Introduzione al corso

Packet switching vs circuit switching

Ci sono due modalità principali per trasferire dati:

- circuit switching: ogni comunicazione ha un circuito dedicato *virtuale* (rete telefonica "vecchia")
- packet switching: i dati sono suddivisi in parti (pacchetti) e vengono inviati attraverso la rete => la base di internet

Circuit switching

Il problema da affrontare nel circuit switching è quello di allocare tutte le risorse (di collegamento e di switching) prima di trasmettere. Oltretutto e2e.

Questo tipo di topologia ha dei problemi:

- le risorse non sono condivisibili
- bisogna eseguire un setup ogni volta che si instaura una connessione

Un grande vantaggio di questo modo di commutazione è la **garanzia delle risorse riservate** e anche della **immunità alle interferenze** da parte di altre comunicazioni.

Sebbene il collegamento sia *virtuale*, ciò che viene fatto nel circuit switching è la allocazione di un dato canale di comunicazione su un mezzo fisico per un dato periodo di tempo. Ogni x periodi alla portante di un determinato canale viene moltiplicata la modulante che trasporta il segnale.

Pertanto il circuit switching si basa sul **multiplexing** di un determinato canale.

Tale multiplexing (del circuit switching) è deterministico, poiché viene dato "a turno" il controllo del canale.

Il multiplexing del circuit switching può essere:

- time division
- frequency division

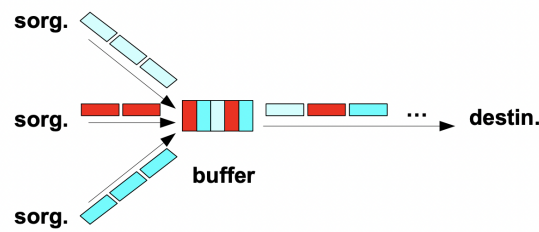
Packet switching

Con il packet switching cade l'idea della **riservatezza delle risorse**. Ogni comunicazione è suddivisa in pacchetti che condivideranno quindi le risorse della rete. Se il canale è libero i pacchetti vengono inviati.

Con il packet switching il canale viene usato **al bisogno**, usando tutta la capacità del link. Questo complica di molto il tipo di multiplexing che il canale deve effettuare, che diventa **statistico**.

All'interno del pacchetto dovranno esserci eventuali informazioni che riconducono il pacchetto ad un determinato tipo di comunicazione.

- **Si bufferizzano i pacchetti in conflitto per lo stesso link**
 - **Il buffer determina in pratica una coda di pacchetti che può essere processata in ordine FIFO (First-In-First-Out), ma non necessariamente (es., in base alla priorità)**
- **Congestione = riempimento del buffer**



Un tipo di comunicazione di packet switching può essere la **store and forward**.

Per ogni hop (router), i pacchetti vengono accodati e inviati verso l'hop successivo attraverso un determinato canale trasmissivo quando quest'ultimo sarà disponibile.

Nella comunicazione store-and-forward ogni pacchetto è indipendente, pertanto pacchetti che si riferiscono alla stessa comunicazione possono intraprendere percorsi diversi: l'importante è che arrivino tutti alla stessa destinazione, ovviamente.

È possibile infatti che si verifichi una **congestione** su particolari nodi della rete:

- la domanda aggregata di risorse può eccedere la quantità di risorse che il nodo può garantire;
- i pacchetti si accodano e quando la coda è piena i **pacchetti vengono persi**.

Pertanto è necessario ridurre la probabilità di **perdita di pacchetti** al minimo possibile.

Ricapitolando, il **packet switching**:

PRO:

- C'è condivisione di risorse
- Non c'è la necessità di prenotare risorse end-to-end
- Il packet switching è ottimo per dati che arrivano in gruppi

CONTRO: Rischi di congestione:

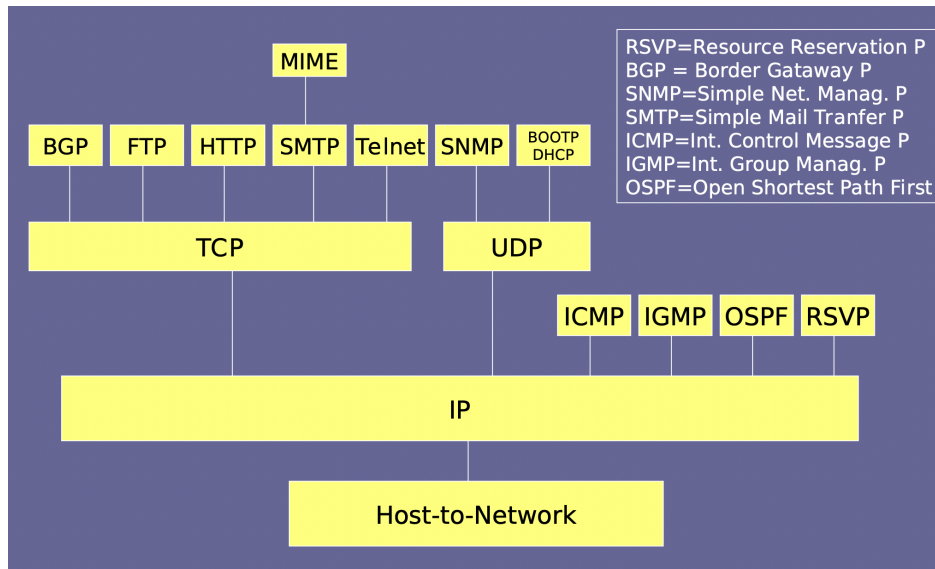
- Ritardo e perdita di pacchetti
- E' necessario un protocollo che garantisca almeno le seguenti due proprietà:
 - Trasferimento dei dati affidabile (in grado di capire se c'è perdita di pacchetti e in grado di provvedere)
 - Controllo della congestione

In particolare, data una connessione a 1Mbit/s, immaginiamo di avere una richiesta da parte di ciascun utente di 0.1 Mbit/s per il 10% del tempo.

Con il circuit switching posso avere **solo** 10 connessioni, sprecando il 90% delle risorse.

Con il packet switching, invece, devo considerare quale sia la probabilità di trovare una congestione (senza buffer). Si ha congestione quando (sempre senza buffer) si hanno 10 utenti che condividono contemporaneamente. Con 35 utenti, la probabilità che più di 10 utenti trasmettano contemporaneamente è molto bassa. Pertanto è possibile che 35 utenti possano usare la stessa linea con rischi minimi.

Stack TCP/IP



Notiamo come la forma dello stack sia “a clessidra”, nel senso che esistono tantissimi protocolli ad alto livello che si vanno a “restringere” all’interno di IP, che è ovviamente il protocollo utilizzato per tutto internet.

Sotto il protocollo IP, i protocolli che comunicano **tra** elementi dello stesso network si diramano di nuovo.