

Introduzione alla commutazione

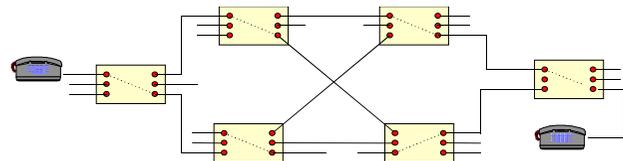
- ✓ Le reti di telecomunicazioni possono essere classificate in:
 - reti a commutazione
 - fanno uso di nodi di commutazione interni
 - ogni nodo di commutazione inoltra il segnale dalla sorgente verso il destinatario (*routing*)
 - reti broadcast
 - non fanno uso di nodi di commutazione
 - il segnale viene ricevuto da tutte le stazioni connesse alla rete
 - LAN, MAN

Introduzione alla commutazione

- ✓ La **commutazione** è una fase fondamentale nella trasmissione dei dati sulla rete
- ✓ Permette di stabilire un percorso fisico fra la sorgente e la destinazione dell'informazione
- ✓ Consente l'uso delle risorse condivise della rete
- ✓ Esistono tre diverse tipologie di commutazione:
 - commutazione di circuito
 - commutazione di pacchetto
 - tecniche ibride

Commutazione di circuito

- ✓ Alla richiesta di connessione da parte di un utente si instaura un percorso fisico **dedicato** fra le estremità comunicanti che viene mantenuto per tutta la durata delle trasmissioni dei dati.



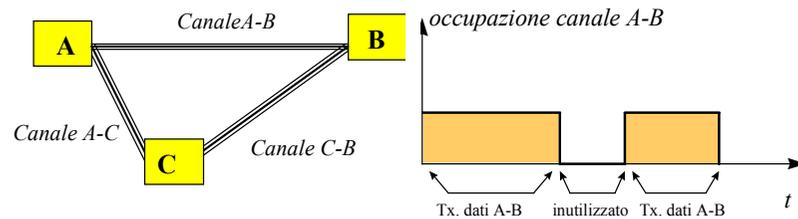
- ✓ Il percorso dedicato agli utenti che ne fanno richiesta consiste in una serie di mezzi trasmissivi interconnessi da nodi di commutazione intermedi.

Tipologie di commutazione

- ✓ I mezzi trasmissivi commutati (*link*) possono essere:
 - canali interamente dedicati agli utenti della comunicazione
 - locazioni (*slot*) temporali in sistemi a divisione di tempo (*TDM*)
 - locazioni in frequenza in sistemi a divisione di frequenza (*FDM*)
- ✓ L'esistenza di un percorso fisico dedicato non implica necessariamente che le risorse utilizzate per realizzare tale percorso non possano essere condivise
- ✓ In pratica la commutazione di circuito garantisce agli utenti di ogni connessione una *risorsa dedicata virtuale* disponibile per il tempo richiesto, sul percorso stabilito

Canale dedicato

- ✓ **Canale interamente dedicato:** la banda del canale di trasmissione è interamente riservata ad una comunicazione, indipendentemente dal fatto che la risorsa sia adeguatamente sfruttata o meno.

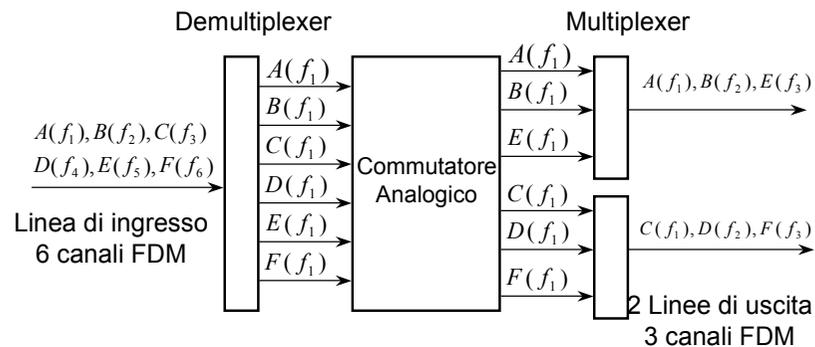


Canale condiviso

- ✓ Si esegue un multiplexing per la trasmissione contemporanea di n segnali in un unico canale.
- ✓ Tipologie di moltiplicazione:
 - Frequency division multiplexing (FDM)
 - trasmettono ogni segnale ad una diversa portante
 - TDM sincrono (Time Division Multiplexing)
 - vengono trasmessi porzioni di segnale intervallati temporalmente
 - gli slot sono preassegnati: ci può essere una perdita di tempo
 - TDM statistico
 - se in un canale in ingresso non c'è segnale il multiplexer passa all'ingresso successivo

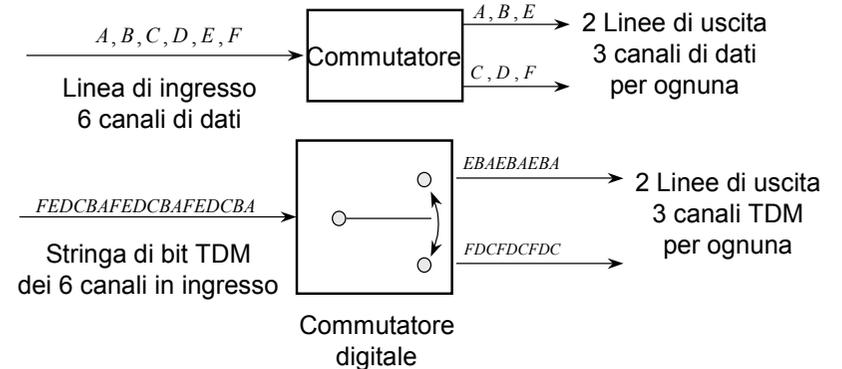
Commutatori analogici (FDM)

- ✓ I commutatori analogici possono essere rappresentati come nello schema seguente:



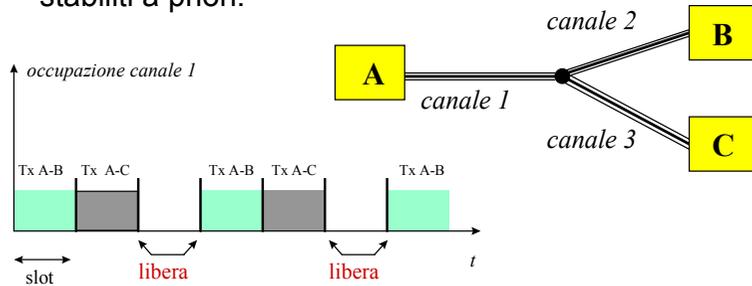
Commutatori digitali (TDM)

- ✓ Completiamo lo schema precedente esemplificando la struttura dei blocchi digitali di commutazione TDM:



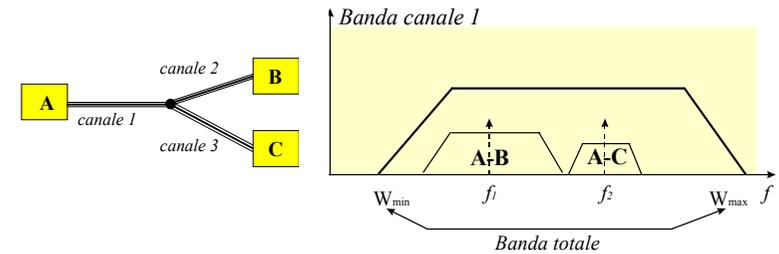
Commutazione di circuito

- ✓ **Canale a divisione di tempo (T.D.M.):** la risorsa viene suddivisa in slot temporali che sono ripartiti fra diverse comunicazioni. La frequenza e il numero degli slot assegnati ai diversi utenti sono fissi e stabiliti a priori.



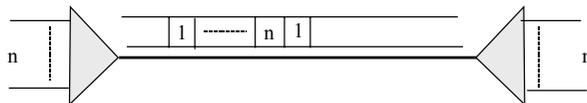
Commutazione di circuito

- ✓ **Canale a divisione di frequenza (F.D.M.):** la banda del canale è suddivisa in porzioni assegnate a utenti diversi. La larghezza di banda associata a ciascun utente e il numero totale di utenti e' fisso e stabilito a priori.



Commutazione basata sul TDM

- ✓ Non esiste una separazione fisica delle connessioni
- ✓ Ogni segnale in ingresso (bitrate R_i) viene diviso in segmenti e trasmesso su un'unica linea in uscita con un bitrate maggiore assieme ad altri n segnali (bitrate in uscita $\geq n \times R_i$)
- ✓ Si fa' uso del *Time Division Multiplexing*

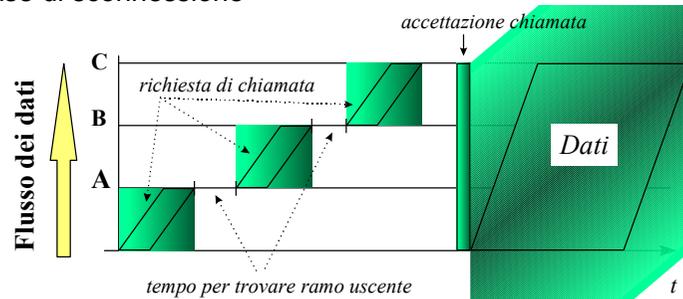


Proprietà della commutazione di circuito

- ✓ La commutazione di circuito ha le seguenti importanti proprietà:
 - il percorso da estremità a estremità deve essere stabilito prima del trasferimento dei dati;
 - prima che possa iniziare la trasmissione dei dati, un segnale di richiesta di chiamata deve propagarsi lungo tutto il percorso fino a destinazione;
 - dopo che la connessione è stata instaurata, l'unico ritardo per i dati è quello dovuto al tempo di propagazione dei segnali sui canali di trasmissione → non esiste il rischio di congestione.

Commutazione di circuito

- ✓ Fasi fondamentali:
 - fase di connessione
 - fase di trasmissione
 - fase di sconnessione



Commutazione di circuito

- **Fase di connessione:** inoltra la richiesta di connessione dall'utente chiamante al nodo di commutazione più vicino e, successivamente, attraverso un percorso libero, all'utente di destinazione. Tale utente, a sua volta, deve inviare un segnale di accettazione (*ACK*) o non accettazione (*BUSY*) della chiamata.
- ✓ **Esempio:** Nella chiamate telefoniche la fase di connessione corrisponde all'intervallo di tempo che intercorre da quando l'utente chiamante digita il numero a quando inizia il segnale di *libero* o di *occupato*. Nel primo caso (*libero*) l'utente destinatario accetta la connessione, nel secondo, invece, è inviato un segnale di *BUSY* (*occupato*).

Commutazione di circuito

- **Fase di trasmissione:**
 - parte di inoltra effettivo dell'informazione
 - ha durata variabile dipendente sia dalla lunghezza dei dati da trasmettere che dalle caratteristiche delle risorse di comunicazione
 - fattori significativi per il ritardo medio dei dati sono la velocità di trasmissione, il numero dei canali disponibili, la distanza da percorrere ...
- ✓ **Fase di sconnessione:**
 - la connessione viene dichiarata conclusa: viene inviato un segnale di controllo che si propaga da un utente agli interlocutori e ai gestori della rete con il quale viene rilasciato e dichiarato disponibile.

Osservazioni

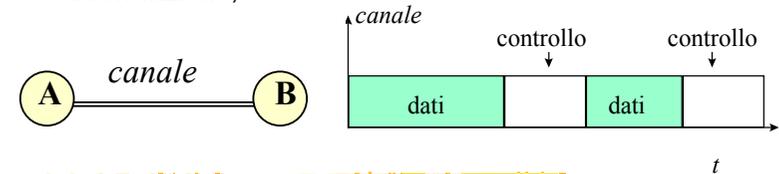
- ✓ Le tre fasi in cui si articola ogni connessione gestita a commutazione di circuito sono, nel loro complesso, abbastanza onerose dal punto di vista:
 - del tempo richiesto per effettuare il collegamento;
 - dei segnali di controllo necessari per propagare tutte le informazioni di controllo all'atto della richiesta di chiamata e del rilascio.
- ⇒ La commutazione di circuito è **vantaggiosa** in tutti quei casi in cui la fase di trasmissione è considerevolmente più lunga delle fasi di connessione e sconnessione, ovvero quando il rapporto fra il tempo effettivo di utilizzo (*trasmissione*) e il tempo totale di occupazione della rete è sufficientemente alto.

Esempio: traffico telefonico e traffico di dati

- ✓ Nel caso di comunicazioni fra calcolatori digitali, caratterizzate da traffico intermittente (trasmissioni brevi, ma frequenti), la tecnica basata sulla commutazione di circuito non sarebbe vantaggiosa, perché il tempo necessario a stabilire la connessione è più lungo del tempo di trasmissione dei dati stessi e le continue richieste indurrebbero ad una mole smisurata di segnali di controllo → spreco di risorse e basse prestazioni.
- ✓ Nel caso di chiamate telefoniche, per le caratteristiche della connessione stessa (conversazioni lunghe e relativamente poco frequenti nel tempo), la commutazione di circuito risulta particolarmente adatta.

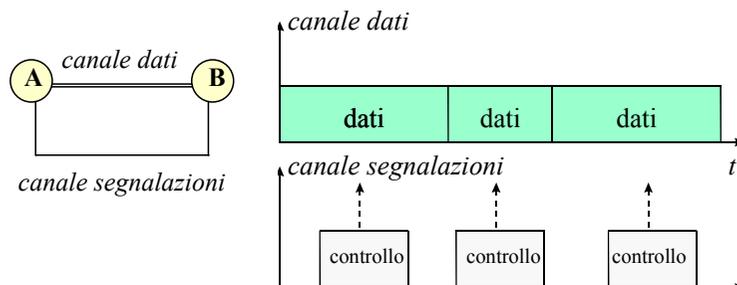
Segnalazione

- ✓ In alcuni casi, per ovviare all'inconveniente del sovraccarico dovuto ai segnali di controllo, si utilizzano canali dedicati ai soli segnali di controllo, in modo che possano viaggiare parallelamente ai dati.
- ✓ Sulla base di queste considerazioni si distingue la: *segnalazione in banda* e *segnalazione a canale comune*:
 - **segnalazione in banda**: i messaggi di segnalazione viaggiano sugli stessi canali dei messaggi di informazione;



Segnalazione

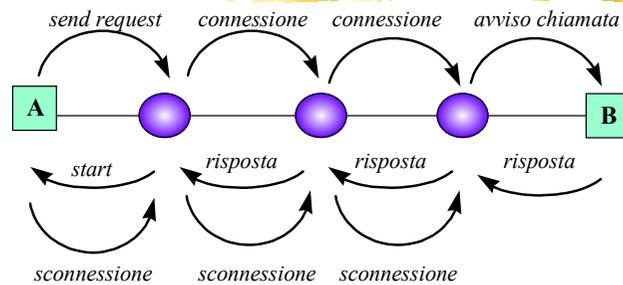
- **Segnalazione a canale comune**: i messaggi di segnalazione viaggiano su canali separati, parallelamente al traffico dei dati.



Analisi di una connessione a CC

- ✓ L'analisi di una connessione gestita a commutazione di circuito ha i seguenti obiettivi principali:
 - studio dettagliato delle varie fasi in cui si articola la connessione;
 - stima dei **tempi medi** necessari a ciascuna fase;
 - stima del **tempo totale medio di occupazione** della rete da parte di una chiamata fra utenti.
- ✓ Per semplificare l'analisi si studia il caso della chiamata fra due soli utenti e si utilizza la **teoria delle code**.
- ✓ Il percorso ideale fra due utenti in una rete a commutazione di circuito può essere rappresentato mediante il seguente diagramma:

Connessione a commutazione di circuito



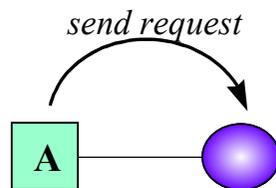
- ✓ In esso si possono evidenziare:
 - i due utenti che partecipano alla connessione □
 - i nodi del percorso fisico che collega i due utenti ○
 - i link o canali di trasmissione, che collegano i nodi —

Tempo di connessione: *call time*

- ✓ La durata della connessione fra due utenti prende il nome di "**call time**" e prevede le seguenti operazioni:
 - 1. chiamata
 - 2. risposta
 - 3. trasmissione
 - 4. sconnessione
- ✓ Le operazioni 1 e 2 fanno parte della procedura di "**call set up**" e le restanti costituiscono la procedura di "**holding**".

La chiamata

- ✓ L'utente A (sorgente), che vuole stabilire una connessione con l'utente B (destinazione), genera una **richiesta di trasmissione**, mediante l'apposito segnale di controllo inviato al nodo di commutazione più vicino.

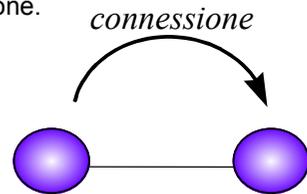


La chiamata

- ✓ Tale segnale (*send request*) contiene:
 - l'indirizzo dell'utente con cui si vuole stabilire la comunicazione
 - altre informazioni di controllo necessarie ai gestori della rete e ai nodi di commutazione del percorso
- ✓ Esempio (chiamata telefonica):
 - questa fase rappresenta il cambiamento dello stato dell'apparato chiamante (da libero ad occupato) e la composizione del numero di identificazione dell'utente richiesto. Le informazioni del segnale di richiesta sono rappresentate perciò dal numero digitato e da altre informazioni aggiuntive quali, ad esempio, la frequenza della portante del segnale.

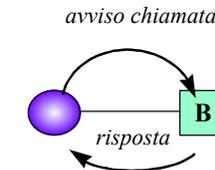
La chiamata

- ✓ Il nodo di commutazione che riceve il segnale di richiesta esegue le seguenti operazioni:
 - elabora ed interpreta le informazioni contenute nella *send request*;
 - seleziona il canale trasmissivo più appropriato per la richiesta (la cosiddetta funzione di *routing*);
 - invia un segnale di connessione al successivo nodo previsto nel percorso, in modo tale che la richiesta venga inoltrata fino all'utente di destinazione.



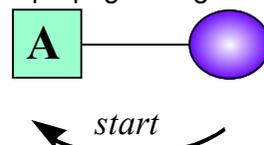
La chiamata

- ✓ L'ultimo nodo del percorso all'arrivo del segnale di richiesta di connessione informa l'utente B della chiamata in arrivo mediante un opportuno segnale (*drilling*):
 - se l'utente B è disponibile accetta la chiamata ed invia un segnale di risposta positiva (*answer*);
 - se l'utente B non è disponibile la richiesta viene rifiutata e viene inviato un segnale di occupato (*busy*).



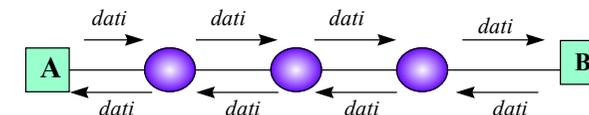
La risposta

- ✓ Questa fase rappresenta la propagazione della risposta dell'utente chiamato verso il chiamante, sia essa positiva o negativa. Essa prevede le seguenti operazioni:
 - il segnale di risposta viene propagato all'indietro attraverso i nodi del percorso stabilito;
 - in caso affermativo il nodo comunicante con l'utente A invia un segnale di inizio trasmissione (*start to send*), in caso negativo si propaga il segnale di occupato (*busy*).



La trasmissione

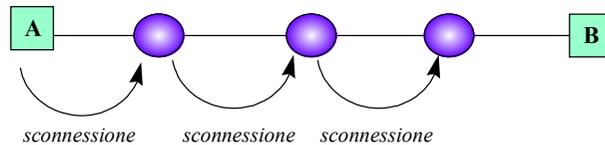
- ✓ Durante questa fase vengono propagati lungo il percorso stabilito i dati dei due utenti.



- ✓ Le modalità della comunicazione sono quelle imposte dalle convenzioni stabilite a priori dagli utenti con il gestore della rete (*half, full duplex o simplex ..ecc*).
- ✓ La presenza di una linea di trasmissione completamente dedicata ai terminali comunicanti fa sì che durante la fase di comunicazione vera e propria non si debbano inviare segnali di controllo: la linea selezionata è infatti mantenuta occupata per tutta la durata della chiamata.

La sconnessione

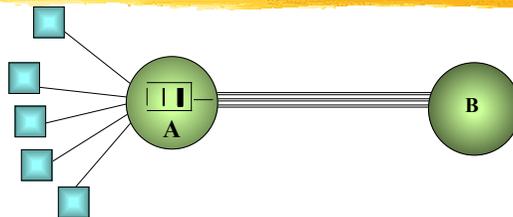
- ✓ Quando uno dei due utenti decide di terminare la comunicazione inizia la fase di sconnessione:
 - l'utente che decide di concludere la comunicazione invia un segnale di fine connessione (*clear down*) al nodo più vicino;
 - il primo nodo propaga tale segnale ai nodi successivi sull'intero percorso;
 - il percorso stabilito fra gli utenti A e B viene rilasciato e dichiarato libero.



Stima del tempo medio di *set-up*

- ✓ Il tempo necessario a stabilire la connessione in una rete a commutazione di circuito è un parametro molto importante, perché influisce pesantemente sulle prestazioni della rete stessa.
- ✓ Tanto più lento è il sistema ad offrire il servizio, tanto meno soddisfacente è la qualità del servizio agli utenti e minore il numero di utenti che si possono servire in un intervallo di tempo fissato.
- ✓ È quindi necessario avere una buona stima di questo parametro che evidenzia, in modo particolare, quali fasi hanno maggiore responsabilità nel determinarne il valore.
- ✓ Per effettuare tale stima in maniera semplice, verrà utilizzato un modello di rete semplificata.

Stima del tempo medio di *set-up*



- ✓ In particolare:
 - il nodo A è un nodo terminale della rete (comunica con l'esterno e si collega alle interfacce degli utenti);
 - il nodo B è un nodo intermedio ed è connesso mediante N linee di trasmissione ai nodi limitrofi (possono essere N canali distinti oppure un unico canale condiviso fra N chiamate);

Stima del tempo medio di *set-up*

- per modellare matematicamente una rete a commutazione di circuito mediante la teoria delle code, si considera ogni singolo nodo di commutazione come una code che inoltra i dati in ingresso se o quando trova un canale di tx libero;
 - generalmente, nelle reti reali, tali code sono bloccanti, perché i dati non vengono inoltrati se non esiste un percorso disponibile all'atto della richiesta
 - tuttavia, per semplificare l'analisi, verranno considerate inizialmente delle code non bloccanti di tipo *store and forward*, in grado cioè di memorizzare i dati in ingresso ed inviarli al momento opportuno.
- per semplicità si supponga che la modalità di trasmissione implementata dalla rete sia **in-band** (vengono trasmessi sul canale sia i dati utente che quelli di controllo).

Tempo di *set-up*

- ✓ Il tempo di *set-up* fra i due nodi generici A e B, nelle ipotesi suddette, si può rappresentare nel seguente modo:



Tempo di *set-up*

- ✓ Analizziamo i vari termini nel dettaglio:
 - T_{SR} (**tempo per inviare la richiesta di connessione**): e' il tempo necessario all'utente per segnalare al nodo più vicino la richiesta di connessione.
 - Ipotesi: nella realtà tale quantità non e' costante; tuttavia, nella presente analisi si suppone che tutti i tempi di segnalazione siano fissi e pari a T_S .
 - $E\{T_p\}$ (**tempo medio di elaborazione della richiesta**): e' il tempo necessario ai nodi per elaborare i messaggi di segnalazione.
 - Ipotesi: si ipotizza per semplicità che i tempi di elaborazione siano uguali per tutti i nodi e pari a $E\{T_p\}$.

Tempo di *set-up*

- $E\{W_p\}$ (**tempo medio di attesa in coda**): e' il tempo medio di attesa nella coda del nodo A, prima che uno degli N canali previsti sia libero per inoltrare la richiesta.
 - Ipotesi: tale quantità e' per sua definizione aleatoria e verrà studiata mediante la teoria delle code non bloccanti.
- T_I (**tempo di propagazione della richiesta**): e' il tempo necessario ai nodi per propagare la richiesta di connessione dall'utente sorgente a quello di destinazione.
 - Ipotesi: tale quantità dipende da molti fattori, fra i quali la lunghezza del percorso e il numero dei nodi della rete; tuttavia per semplicità si assume fissa e pari a T_I .

Tempo di *set-up*

- T_R (**tempo di risposta**): è il tempo impiegato dall'utente destinazione per rispondere positivamente o negativamente alla richiesta di chiamata.
 - Ipotesi: essendo il tempo richiesto per un messaggio di segnalazione viene assunto fisso e pari a T_S .
- T_S (**tempo di inizio trasmissione**): e' il tempo impiegato dall'utente sorgente per organizzare la trasmissione dei dati dopo aver ricevuto l'autorizzazione alla comunicazione.
 - Ipotesi: Essendo anch'esso il tempo richiesto per un messaggio di segnalazione viene assunto fisso e pari a T_S .

Tempo di *set-up*

- ✓ In conclusione, dopo le ipotesi semplificative effettuate per l'analisi, il tempo di *set-up* complessivo T_C risulta così composto:

$$T_C = 3T_S + T_I + 3E\{T_P\} + E\{W\}$$

quantità deterministiche *quantità aleatorie*

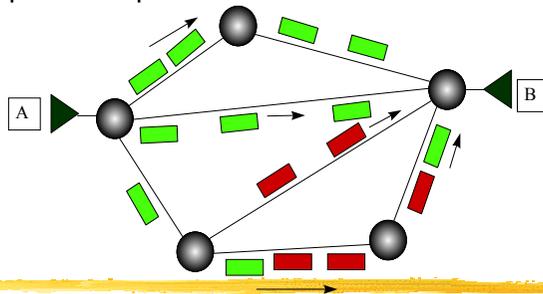
- ✓ Le quantità aleatorie sono difficili da valutare, ma al tempo stesso risultano particolarmente influenti per le prestazioni del sistema.

Digital PBX

- ✓ PBX (Digital Branch Exchange)
 - rete per la connessione di più terminali, telefonici all'interno di una struttura privata
 - fino agli anni venti la commutazione era manuale e poi si è passati a dei dispositivi elettromagnetici
 - dopo gli anni settanta si è passati ai digital PBX

Commutazione di pacchetto

- ✓ I dati di messaggio sono incapsulati in blocchi di lunghezza massima fissata e inviati sulla rete in modo indipendente:
 - in pratica, ogni blocco di dati è una entità autonoma in grado di raggiungere la propria destinazione attraverso un percorso qualsiasi fra i nodi della rete.



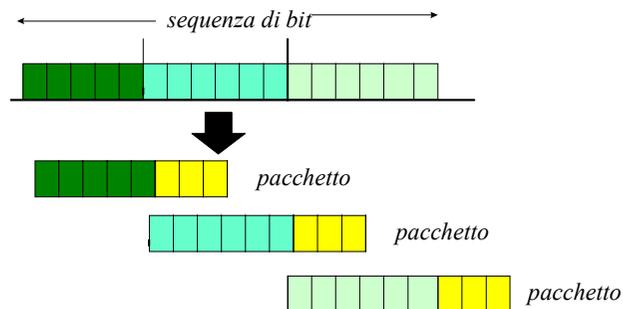
I pacchetti

- ✓ In ogni blocco di dati vengono inserite delle informazioni aggiuntive ("overhead"), contenenti tutti i dati necessari affinché il blocco raggiunga la destinazione finale e affinché il messaggio possa essere ricostruito dall'utente ricevente.
- ✓ L'insieme dei bit di informazione e dei bit di *overhead* prende il nome di **pacchetto**.



- ✓ I pacchetti sono generati o dagli stessi terminali utente o dall'interfaccia rete-utente la quale suddivide i dati di messaggio che le arrivano e vi aggiunge le informazioni di sistema necessarie.

I pacchetti



- All'interno della rete non viene stabilito per i pacchetti un percorso fisico dedicato fra utente sorgente e utente destinazione, ma i pacchetti viaggiano di nodo in nodo sulla base delle risorse disponibili al momento.

Caratteristiche del pacchetto

- ✓ Le caratteristiche dei pacchetti (lunghezza dei dati, numero di campi ..ecc) possono variare ampiamente con il tipo di rete.
- ✓ In generale, tuttavia, essi devono sempre contenere le seguenti informazioni:
 - dati di messaggio;
 - indirizzo dell'utente destinazione;
 - indirizzo dell'utente sorgente;
 - informazioni sulla lunghezza del messaggio;
 - numero di sequenza del pacchetto;
 - bit di rivelazione degli errori.

Campi del pacchetto

- ✓ **Dati di messaggio:** hanno lunghezza massima stabilita sulla base delle convenzioni (protocolli) della rete stessa. E' importante osservare che i bit di messaggio dopo essere stati suddivisi in blocchi, perdono qualsiasi significato semantico.
- ✓ **Indirizzo dell'utente di destinazione:** è necessario affinché i nodi della rete possano indirizzare ogni pacchetto lungo un percorso valido per la destinazione indicata. Esso, inoltre, serve all'utente destinazione per riconoscere i pacchetti che gli sono stati indirizzati.

Campi del pacchetto

- ✓ **Indirizzo dell'utente sorgente:** presente in ogni pacchetto è indispensabile al ricevitore per distinguere i pacchetti che giungono da sorgenti differenti.
- ✓ **Lunghezza del messaggio:** è indicativa della lunghezza della parte dati del pacchetto. E' utile nel caso di pacchetti di lunghezza variabile.
- ✓ **Numero di sequenza del pacchetto:** serve al ricevitore per ordinare i pacchetti appartenenti allo stesso messaggio e creare così una sequenza logica di bit di messaggio. Se infatti i pacchetti seguono percorsi diversi per raggiungere la destinazione possono arrivare all'utente finale con un ordine diverso da quello con cui sono partiti.

Bit per la rivelazione degli errori

- ✓ Servono per rivelare eventuali errori di trasmissione che possono alterare il contenuto dei pacchetti durante l'attraversamento della rete di comunicazione
 - tale controllo è solitamente effettuato a livello di nodi intermedi, in modo che la corretta trasmissione nodo-nodo sia sempre garantita;
 - i codici a rivelazione di errore più semplici sono i **codici a controllo di parità**, che permettono di rivelare la presenza di un errore o di un numero dispari di errori di trasmissione in un pacchetto con un solo *bit* aggiuntivo;
 - talvolta, per dati particolarmente vulnerabili, si possono utilizzare codici di rivelazione di errore più complessi, a scapito però della maggiore lunghezza del pacchetto.

Caratteristiche della rete a CP

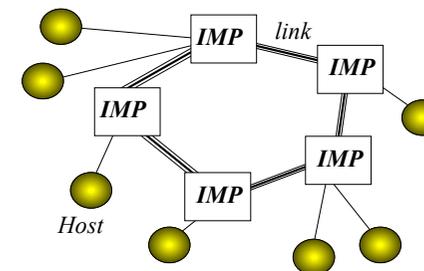
- ✓ Gli elementi che compongono una rete gestita a commutazione di pacchetto sono, come per le reti a commutazione di circuito, i nodi e i link. Sono tuttavia nuove le funzioni che essi svolgono:
 - **nodi**: sono gli elementi di connessione della rete ed hanno la funzione di memorizzare i pacchetti in arrivo e di instradarli fino a destinazione su un generico percorso, implementando una ben precisa *politica di instradamento*;
 - **link**: sono i canali fisici di tx sui quali i pacchetti possono viaggiare. A differenza della commutazione di circuito, ad una tx non viene assegnata una porzione fissa di canale (slot temporale o una banda).

La condivisione delle risorse

- ✓ Sebbene i circuiti a commutazione di pacchetto non prevedano tecniche di moltiplicazione per la suddivisione dei canali fra più comunicazioni, essi permettono tuttavia la condivisione delle risorse.
- ✓ I *link* che connettono i nodi di un percorso, infatti, possono essere attraversati da pacchetti che appartengono a utenti diversi e che condividono lo stesso tratto di percorso.
- ✓ L'identificatore di pacchetto contenuto nell'*overhead* serve appunto per individuare in modo univoco i pacchetti appartenenti alle rispettive comunicazioni.

Esempio

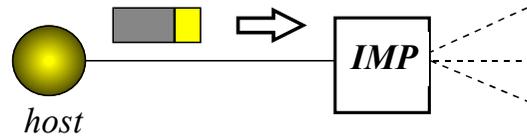
- ✓ Per chiarire il meccanismo di trasmissione della commutazione di pacchetto si consideri come esempio la rete *ARPANET*: essa è uno dei primi modelli di rete a pacchetto su larga scala e può essere così rappresentata:



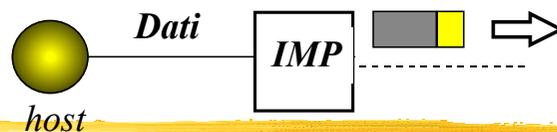
gli **IMP** (*interface message processors*) sono i nodi della rete; i **link** sono cavi full-duplex a 50 kbps; gli **host** sono gli utenti della rete.

Inoltro dei pacchetti

- Gli utenti (*host*) che vogliono comunicare con un utente della rete preparano i pacchetti e li inviano al nodo di interfaccia con la rete.

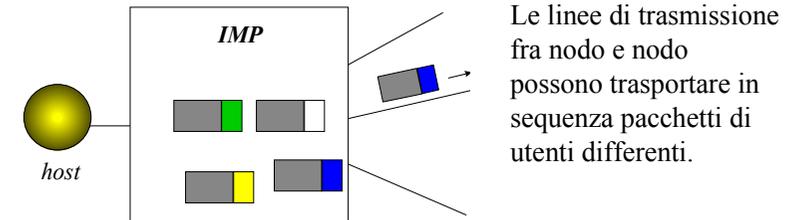


- In alternativa, se i terminali utenti sono di piccola entità, sono le stesse interfacce (*IMP*) ad occuparsi dell'impacchettamento dei dati.



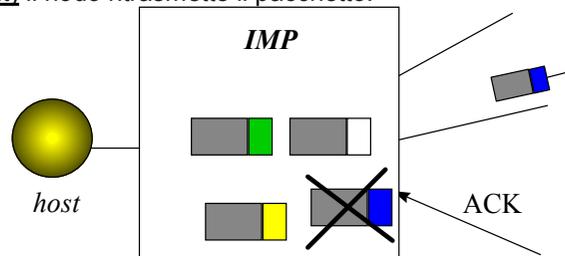
Inoltro dei pacchetti

- I pacchetti attraversano la rete in modalità **store and forward**: essi sono infatti memorizzati all'interno di ogni nodo in attesa di una risorsa libera.
- Il nodo determina l'itinerario del pacchetto e gli assegna la risorsa appropriata appena questa diventa disponibile.



Inoltro dei pacchetti

- Quando un *IMP* trasmette un pacchetto al nodo successivo rimane in attesa di un segnale di risposta:
 - se gli giunge un segnale di *ACK* può cancellare il pacchetto dalla propria memoria, in quanto il segnale di *ACK* conferma che il pacchetto è stato correttamente trasmesso;
 - se non arriva alcun segnale entro un tempo prefissato (*time-out*) il nodo ritrasmette il pacchetto.



Inoltro dei pacchetti

- ✓ A sua volta, quando un nodo intermedio riceve un pacchetto, invia il segnale di *ACK* se sono verificate le seguenti due condizioni:
 - il pacchetto è stato trasmesso correttamente (il controllo di parità fornisce zero errori);
 - il nodo può ospitare il pacchetto nel proprio *buffer*.
- ✓ I nodi di destinazione, infine, ricompongono i pacchetti in un unico messaggio e inviano segnalazioni di *ACK* all'utente sorgente per confermare la corretta ricezione del messaggio.

Osservazioni

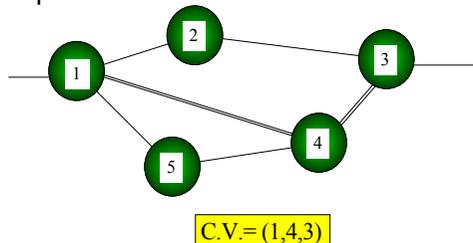
- ✓ Nei termini della standardizzazione OSI, le procedure indicate nell'esempio appartengono a livelli di astrazione differenti:
 - la fase di propagazione del pacchetto da un nodo all'altro nel suo percorso fino a destinazione è un esempio di protocollo del **livello di link** (livello 2);
 - all'interno di ogni nodo, la fase di determinazione dell'itinerario del pacchetto corrisponde ad una procedura del **livello di rete** (livello 3);
 - le fasi di ricomposizione del messaggio nel nodo terminale e di segnalazione finale (*end to end*) corrispondono a procedure del **livello di trasporto** (livello 4).

Proprietà della commutazione di pacchetto

- ✓ Note le caratteristiche generali della commutazione di pacchetto, si possono ora distinguere due diversi modi di trasmissione: **Circuito Virtuale** e **Datagram**.
 - **Circuito Virtuale**: realizza un circuito logico fra mittente e destinatario sul quale è inviato in maniera ordinata l'intero flusso di dati della comunicazione (trasmissione orientata alla connessione);
 - **Datagram**: non esiste un circuito logico fissato fra mittente e destinatario, ma i pacchetti del flusso di dati della comunicazione sono inviati sulla rete in modo totalmente indipendente (trasmissione senza connessione).

Trasmissione a Circuito Virtuale

- ✓ Prima dell'inizio della trasmissione viene realizzato un circuito logico fra utente sorgente e utente destinazione attraverso cui passano **tutti** i pacchetti dell'intero messaggio:
 - **circuito logico**: sequenza di nodi che identifica un certo percorso fra i rami della rete.



Trasmissione a Circuito Virtuale

- ✓ Tutti i pacchetti di ogni singola connessione effettuano il medesimo percorso prima di giungere a destinazione
 - la rete consegna al destinatario i pacchetti in **sequenza ordinata**.
- ✓ In realtà esistono diverse situazioni in cui i pacchetti, pur essendo trasmessi in sequenza ordinati giungono a destinazione disordinati. Ciò può essere causato dalla perdita di alcuni pacchetti (o ritrasmissione):
 - ad un certo livello nella stazione ricevente si dovrà effettuare un controllo ed eseguire il riordinamento dei pacchetti in modo che i livelli superiori vedano la connessione come un **canale ideale**.

Trasmissione a Circuito Virtuale

✓ Analogamente a quanto si verifica nelle reti a commutazione di circuito, la trasmissione a Circuito Virtuale si articola nelle seguenti fasi fondamentali: connessione, trasmissione, sconnessione.

➤ **Fase di connessione:**

- si stabilisce un percorso logico fra sorgente e destinazione;
- si definisce un identificatore per il Circuito Virtuale;
- si aggiornano tutti i nodi del percorso sulle regole di instradamento con cui vanno inoltrati i pacchetti del Circuito Virtuale così definito.

➤ **Fase di trasmissione:**

- i pacchetti sono propagati sulla rete in sequenza.

Trasmissione a Circuito Virtuale

➤ **Fase di sconnessione:**

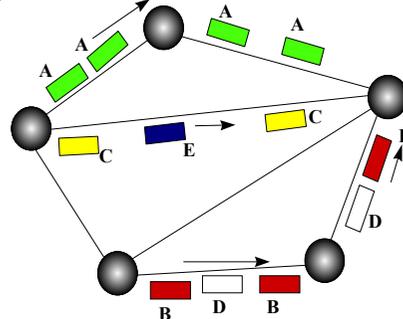
- il messaggio viene ricostruito dal ricevitore;
 - viene rilasciato il percorso logico fra sorgente e destinazione;
 - si aggiorna lo stato (*disponibile*) di tutti i nodi situati sul percorso del Circuito Virtuale.
- ✓ Le tre fasi indicate non si riferiscono ad un circuito fisico dedicato alla singola comunicazione, ma ad un Circuito Virtuale condiviso:
- più circuiti virtuali possono infatti condividere lo stesso percorso!

Condivisione delle risorse nei CV

✓ La caratteristica principale della trasmissione a Circuito Virtuale è quella di simulare un canale fisico dedicato agli utenti, usufruendo, al tempo stesso, dei vantaggi della commutazione di pacchetto.

✓ Il primo di questi vantaggi è la condivisione delle risorse.

I pacchetti appartenenti a Circuiti Virtuali (CV) diversi possono passare attraverso gli stessi canali di trasmissione, permettendo il massimo sfruttamento della risorsa.



Il problema dell'identificazione nei CV

✓ La necessità di assegnare a tutti i pacchetti di ogni Circuito Virtuale un **identificatore di CV** scaturisce dalle seguenti osservazioni:

- Fra mittente e destinatario esistono diversi Circuiti Virtuali possibili
- l'indirizzo degli utenti comunicanti non è sufficiente a determinare univocamente un percorso.

La numerazione dei CV

- ✓ L'identificazione dei Circuiti Virtuali avviene tramite **numerazione**.
- ✓ Ad ogni Circuito Virtuale è assegnato un numero di identificazione che è copiato su tutti i pacchetti del flusso di dati.
- ✓ I nodi della rete contengono al loro interno delle **tabelle di corrispondenza** che associano ad ogni identificatore il relativo percorso (p.e. l'indirizzo del nodo successivo).
- ✓ All'arrivo di ogni pacchetto sono quindi eseguite le seguenti operazioni:
 - lettura del numero di identificazione;
 - ricerca sulla tabella del nodo di percorso;
 - instradamento del pacchetto.

La numerazione dei CV

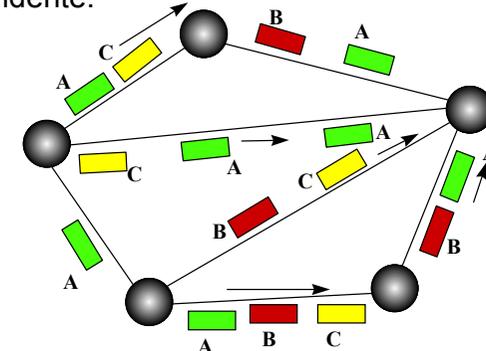
- ✓ Il modo concettualmente più semplice per numerare i Circuiti Virtuali è quello di utilizzare una **numerazione globale** per tutta la rete:
 - poiché il numero di CV di una rete può essere molto elevato, la numerazione globale ha lo svantaggio di richiedere all'interno di ogni pacchetto campi di controllo molto lunghi.
- ✓ Poiché i bit di controllo non apportano informazione utile, i pacchetti con campi di controllo lunghi non sono efficienti:
 - sono state sviluppate delle tecniche di numerazione più articolate, in grado di identificare univocamente i Circuiti Virtuali riducendo al minimo l'**overhead**.

Conclusioni sulla trasmissione a CV

- ✓ I vantaggi principali della trasmissione a Circuito Virtuale sono i seguenti:
 - consegna i pacchetti in **sequenza ordinata**;
 - effettua il **controllo** sui pacchetti trasmessi;
 - fornisce un servizio analogo alla commutazione di circuito, senza richiedere un circuito fisico dedicato → **maggiore flessibilità**
- ✓ Gli svantaggi legati alla trasmissione a circuito virtuale riguardano invece:
 - il **maggiore onere** di lavoro per i nodi della rete;
 - il **ritardo di trasmissione** dovuto alla necessità di trovare un percorso virtuale su cui inoltrare tutti i pacchetti del flusso di dati.

Trasmissione Datagram

- ✓ I pacchetti di un singolo flusso di dati viaggiano sulla rete attraverso percorsi differenti. Ogni pacchetto, detto anche **datagram** è una entità totalmente indipendente.

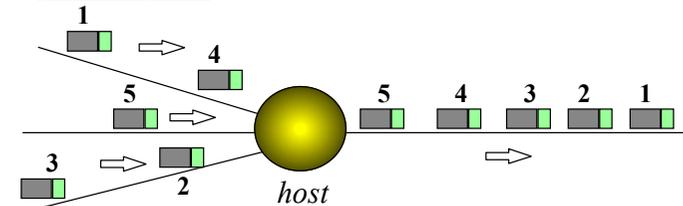


Esempi

- ✓ ARPANET (operativa dal 1969):
 - finanziata inizialmente dall'Advanced Research Project Agency (ARPA) per sperimentazioni
 - dal 1975 diventa una vera rete amministrata dalla Defense Communications Agency
- ✓ TYMNET (operativa dal 1969):
 - originariamente sviluppata per fornire una rete per la connessione di terminali ad una macchina centrale time-sharing a basso costo
- ✓ SNA (dal 1974):
 - non è una vera rete ma è l'approccio più diffuso per la connessione di calcolatori in rete (IBM)

Trasmissione Datagram

- Nella trasmissione **senza connessione** (*connectionless*) non sono previste le fasi di connessione e sconnessione, ma, al contrario, i pacchetti sono instradati sulla base del solo indirizzo di destinazione contenuto nell'overhead:
 - i pacchetti di uno stesso flusso di dati possono raggiungere la destinazione fissata attraverso percorsi differenti → spesso arrivano all'utente finale con un ordine diverso da quello originale.



Trasmissioni Datagram

- ✓ Le reti di tipo *connectionless* risultano più semplici rispetto a quelle orientate alla connessione, poiché il ruolo svolto dai nodi di commutazione è notevolmente semplificato.
- ✓ I nodi della rete, infatti, devono implementare delle regole di instradamento basate sul solo indirizzo di destinazione senza tenere conto della sequenza e dell'integrità dei pacchetti.
- ✓ La rete, infatti:
 - non garantisce consegna ordinata dei pacchetti;
 - non garantisce l'arrivo dei pacchetti a destinazione.

Trasmissione Datagram

- ✓ La semplicità della rete si riflette sull'onere dei terminali di rete o sulle interfacce rete-utente che devono provvedere a:
 - numerare i pacchetti;
 - riordinarli in ricezione;
 - chiedere la ritrasmissione dei pacchetti non pervenuti a destinazione.
- ✓ Se un pacchetto non raggiunge la destinazione entro un tempo limite prefissato, esso viene dichiarato disperso e deve essere ritrasmesso per la corretta ricostruzione del messaggio.

Caratteristiche dei pacchetti

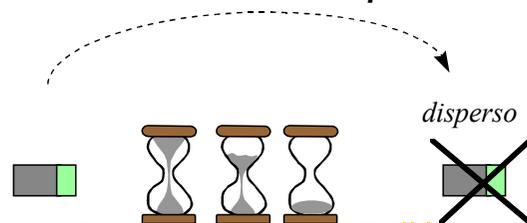
- ✓ I pacchetti di tipo *datagram* devono presentare, fra le informazioni comuni a tutti i pacchetti, due campi specifici: numero di sequenza e indicazione del tempo di vita:
 - **numero di sequenza**: è indispensabile per il riordino dei pacchetti al ricevitore e la ricostruzione del messaggio.
- ✓ Il numero di sequenza, è presente anche nei pacchetti di CV, tuttavia esiste una sostanziale differenza fra il caso di CV ed il caso Datagram: nei CV il numero di sequenza è utilizzato per controllare che tutti i pacchetti siano giunti nell'ordine corretto (se la rete fosse *ideale*, in teoria potrebbe essere omesso); nei circuiti senza connessione, invece, la ricostruzione del messaggio risulterebbe impossibile anche su canali ideali senza l'indicazione del numero di sequenza.

Caratteristiche dei pacchetti

- **indicazione del tempo di vita**: è un parametro che indica il tempo di vita di un pacchetto. Esso è inizializzato all'atto della creazione del pacchetto ed è aggiornato dai nodi della rete durante il tragitto del pacchetto dalla sorgente alla destinazione.
 - Il tempo di vita di un pacchetto può essere calcolato in:
 - **numero di nodi attraversati**: ogni nodo che riceve un pacchetto ne aggiorna il tempo di vita decrementando un contatore;
 - **tempo trascorso dalla creazione**: ogni nodo che riceve il pacchetto ne aggiorna il riferimento temporale mediante una etichetta.

Tempo di vita

- ✓ Il tempo di vita di un pacchetto è un parametro molto importante perché impedisce che un pacchetto possa vagare indefinitamente per la rete senza raggiungere la destinazione.
- ✓ In molti casi, infatti, si stabilisce un tempo di vita massimo per il pacchetto, superato il quale il pacchetto viene considerato **disperso**.



Conclusioni sulla trasmissione Datagram

- ✓ La trasmissione non orientata alla connessione ha i seguenti vantaggi:
 - flessibilità;
 - poco onere di gestione per i nodi della rete.
- ✓ Essa presenta tuttavia i seguenti svantaggi:
 - non consegna i pacchetti in sequenza ordinata → maggiore onere richiesto alle interfacce dei terminali;
 - può causare ritardi nella ricostruzione dei messaggi a causa della perdita dei pacchetti.

Routing (instradamento)

- ✓ Aspetto cruciale nelle reti a CP
- ✓ Attributi di una funzione di routing:
 - correttezza e semplicità;
 - robustezza: scelta di percorsi alternativi;
 - stabilità: talvolta opposta alla robustezza;
 - equità: trattare allo stesso modo tx vicine e distanti.
- ✓ Elementi delle tecniche di instradamento:
 - criteri base:
 - numero di salti, costo, ritardo, throughput
 - momento di decisione:
 - datagram/circuito virtuale

Instradamento

- dove decidere:
 - ad ogni nodo (distribuito), nodo centrale, nodo d'origine
- sorgente di informazione sulla rete:
 - nessuna, locale, da nodi adiacenti, nodi del percorso, tutti i nodi
- strategia di instradamento:
 - fissa, a valanga, random, adattiva
- aggiornamento per il routing adattivo
 - continuo, periodico, cambio topologia o del carico
- ✓ Instradamento fisso:
 - si fa uso di una matrice centrale di routing fissa
 - da questa ogni nodo di commutazione accede per costruirsi la propria tabella di instradamento

Instradamento

- nessuna differenza nel percorso tra datagram e CV
- è semplice e robusta in una rete con carico stabile
- ✓ Instradamento a valanga:
 - ogni pacchetto in ingresso ad un nodo viene instradato su tutti gli altri link
 - al destinatario arrivano più copie dello stesso pacchetto
 - per evitare che pacchetti circolino all'infinito questi vengono eliminati dopo un certo numero di salti
 - robusto (reti militari) ma genera un traffico notevole
- ✓ Instradamento random:
 - ogni nodo inoltra i pacchetti in un altro link in modo random;

Instradamento

- ad ogni link viene assegnata una certa probabilità (p.e. proporzionale al costo del link);
- se le probabilità sono uguali: round robin
- anche in questo caso non c'è bisogno di informazioni sulla rete.
- ✓ Instradamento adattivo:
 - le strategie viste fino adesso non reagiscono alle variazioni della rete o perlomeno reagiscono poco di frequente in seguito ad un'azione di un operatore:
 - potrebbe essere meglio;
 - una strategia adattiva può essere complessa, aumenta il traffico per la tx info sulla rete, può reagire troppo velocemente causando congestioni o troppo lentamente diventando inutile;

Instradamento adattivo

- vantaggi di un routing adattivo: aumento delle performance, può aiutare nel controllo del traffico.
- è in uso nelle reti ARPANET e TYMNET con continue modifiche;
- esistono tre categorie:
 - *routing adattivo isolato*: fa uso di informazioni locali. Il caso più semplice consiste nel routing sulla base delle code su ogni link (facendo magari uso di un bias). Non sono molto usate in quanto fanno uso solo di info locali;
 - *routing adattivo distribuito*: ogni nodo fa delle stime dei ritardi e guasti lungo ogni percorso e fornisce anche queste informazioni ai nodi adiacenti;
 - *routing adattivo centralizzato*: in questo caso queste informazioni vengono inviate ad un nodo centrale che esegue le decisioni di routing e le invia ai nodi.

Esempi

	Original ARPANET	Revised ARPANET	TYMNET I	TYMNET II	SNA
<i>Performance criterion</i>	Delay	Delay	Delay, rate, overload	Data rate, load, link and traffic types	User-defined
<i>Decision Time</i>	Packet	Packet	Session	Session	Session
<i>Decision Place</i>	Each node	Each node	Central node	Central node	Originating node
<i>Information Source</i>	Adjacent nodes	All nodes	All nodes	All nodes	User defined
<i>Routing Strategy</i>	Adaptive	Adaptive	Adaptive	Adaptive	User defined
<i>Update Time</i>	Periodic	Periodic	Load change	Load change	Topology change

Osservazioni sulla CP

- ✓ La commutazione di pacchetto, per le sue caratteristiche, si presta bene alla trasmissione di dati *bursty*:
 - il trasporto di flussi di dati brevi e discontinui nel tempo non comporta spreco di risorse perché i pacchetti viaggiano come entità autonome.
 - la trasmissione a commutazione di pacchetto (sia a Circuito Virtuale che Datagram) è di tipo *store and forward*:
 - ogni pacchetto deve attendere nella coda dei nodi prima di essere inoltrato ⇒ vi è un ritardo aggiuntivo stimabile mediante la teoria delle code.

Osservazioni

- ✓ La trasmissione di pacchetto permette di ottimizzare le risorse:
 - non esiste un circuito fisico fra sorgente e destinazione perciò i pacchetti possono essere inoltrati via via sui percorsi a minore costo (*gestione dinamica della rete*).
- ✓ Tuttavia, si può osservare che:
 - i pacchetti non contengono solo informazione utile, ma hanno il sovraccarico dell'intestazione ⇒ spreco di risorse per trasmettere le informazioni di sistema;
 - la frammentazione dei dati in pacchetti può produrre pacchetti parzialmente vuoti (anche un solo bit di informazione deve essere impacchettato con l'intestazione).

Osservazioni

- ✓ Problemi legati alla lunghezza dei pacchetti:
 - **pacchetti corti:**
 - maggiore flessibilità;
 - minore perdita di tempo nella ritrasmissione di un pacchetto disperso;
 - codici di rivelazione di errore più semplici;
 - basso rapporto informazione - *overhead*.
 - **pacchetti lunghi**
 - minore flessibilità;
 - maggiore perdita di tempo nella ritrasmissione di un pacchetto disperso;
 - codici a rivelazione di errore più complessi;
 - alto rapporto informazione - *overhead*
- Occorre trovare un compromesso fra **flessibilità** ed **efficienza**.

Confronto fra CP e di CC

- ✓ La commutazione di circuito e la commutazione di pacchetto differiscono per molti aspetti che le rendono vantaggiose in casi differenti:
 - la commutazione di circuito riserva **staticamente** in anticipo la larghezza di banda richiesta per la trasmissione;
 - la commutazione di pacchetto acquista e rilascia **dinamicamente** la risorsa quando necessario;
 - nella commutazione di circuito qualunque porzione di banda non utilizzata su un circuito assegnato è semplicemente sprecata;

Confronto

- nella commutazione di pacchetto le porzioni di risorsa non utilizzate da due utenti possono essere utilizzate da altre comunicazioni;
- nella commutazione di circuito non vi sono problemi di congestione del traffico perché quando la risorsa è assegnata **rimane libera** per tutta la durata della trasmissione;
- nella commutazione di pacchetto un improvviso aumento del flusso può saturare le code dei nodi e provocare problemi di **congestione del traffico**;
- nella commutazione di circuito non esiste la fase di riordino perché tutti i dati sono trasmessi come un **unico flusso**;

Confronto

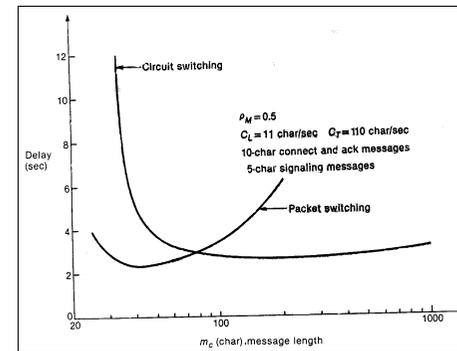
- nella commutazione di pacchetto vi è l'onere del **riordino** e del controllo dei pacchetti;
- nella commutazione di circuito i parametri di trasmissione (velocità, codifica, ecc.) sono **fissi** e **stabiliti a priori**;
- nella commutazione di pacchetto i nodi possono variare **dinamicamente** i parametri di trasmissione sulla base delle caratteristiche della rete → maggiore flessibilità.

Confronto

- nella commutazione di circuito le tariffe da pagare per la trasmissione si basano sulla distanza, sul tempo di occupazione (si pensi alle tariffe telefoniche);
- nella commutazione di pacchetto le tariffe si basano, solitamente, sul numero di *byte* o sul numero di pacchetti trasportati.
- si può dimostrare che la commutazione di circuito fornisce tempi di ritardo inferiori quando i messaggi sono lunghi;
- la commutazione di pacchetto fornisce tempi di ritardo complessivi minori quando la lunghezza dei messaggi è ridotta.

Confronto

- ✓ Per fare un confronto quantitativo fra i due tipi di commutazione si consideri il seguente esempio rappresentativo e le curve ottenute:



esiste un valore di soglia per la lunghezza del messaggio per la quale la commutazione di circuito è da preferirsi alla commutazione di pacchetto; l'una o l'altra tecnica sono da definirsi caso per caso sulla base di stime sui tipi di dati da trasmettere.

Tecniche di commutazione alternative

- ✓ Tecniche di commutazione alternative sono state studiate e sviluppate nel corso degli anni con l'obiettivo di:
 - superare i limiti imposti dalla commutazione di circuito e di pacchetto;
 - rendere le reti sempre più flessibili;
 - ottimizzare al meglio le risorse disponibili;
 - poter trasmettere diverse tipologie di dati contemporaneamente.
- ✓ Fra le tecniche alternative più diffuse ricordiamo: **la commutazione di messaggio** e la **commutazione ibrida**.

Commutazione di messaggio

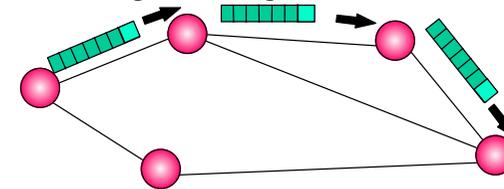
- ✓ La commutazione di messaggio è una variante della commutazione di pacchetto.
- ✓ Infatti:
 - non stabilisce un percorso fisico fra il trasmettitore ed il ricevitore;
 - i dati da trasmettere vengono incapsulati in trame piuttosto lunghe (*messaggi*);
 - i messaggi sono accodati nei *buffer* dei nodi e trasmessi in modalità *store and forward*;
 - ciascun blocco viene ricevuto integralmente, ispezionato e ritrasmesso al nodo successivo.

Differenze fra CP e CM

- ✓ La principale differenza fra la commutazione di pacchetto e la commutazione di messaggio è strettamente connessa alla dimensione dell'unità di dati trasmessi.
- ✓ Nella commutazione di pacchetto il blocco (pacchetto) di dati è relativamente breve;
- ✓ Nella commutazione di messaggio l'unità di informazione è un intero messaggio di definita lunghezza, che occupa il canale di trasmissione per un periodo abbastanza lungo.

Commutazione di messaggio

- ✓ Il meccanismo di trasmissione in modalità a commutazione di messaggio può essere rappresentato mediante il seguente diagramma:



- ✓ Esso raffigura l'andamento temporale di un blocco di dati (messaggio) fra tre nodi consecutivi. Si noti le due fonti di ritardo nella consegna del pacchetto:
 - ritardo di propagazione dei segnali;
 - ritardo di attesa in coda.

Osservazioni

- ✓ A differenza della commutazione di pacchetto, nella commutazione di messaggio, non vi sono limiti alla dimensione dei blocchi di dati da trasmettere:
 - la commutazione di messaggio riduce al minimo l'*overhead* del blocco di dati (1 *overhead* per l'intero messaggio) rendendo la trasmissione più **efficiente**;
 - il rapporto "bit di informazione"/"bit di controllo" è alto;
 - in fase di trasmissione viene sprecato meno tempo per propagare i messaggi di segnalazione (1 segnale per l'intero messaggio).

Osservazioni

- ✓ Per contro, rispetto alla commutazione di pacchetto la commutazione di messaggio presenta i seguenti svantaggi:
 - i buffer dei nodi devono essere sufficientemente grandi da ospitare anche messaggi lunghi in attesa di essere propagati;
 - i canali di trasmissione fra un nodo e l'altro possono rimanere occupati per intervalli di tempo prolungati durante la trasmissione di messaggi lunghi.
- ✓ Questi aspetti fanno sì che in molte applicazioni (ad esempio nella multimedialità) la commutazione di messaggio non sia vantaggiosa, mentre, invece, lo è per traffici continui.

Commutazione ibrida

- ✓ Con il termine *commutazione ibrida* si intendono tutte le versioni avanzate di commutazione di pacchetto e di circuito; fra esse si annoverano la: *connessione di circuito a commutazione rapida* e la *commutazione di pacchetto a divisione di tempo*.
- ✓ **Connessione di circuito a commutazione rapida:** è un tipo di commutazione di circuito che si propone di superarne i limiti del lungo tempo di connessione;
 - e', in pratica, una commutazione di circuito in cui il tempo di connessione è sull'ordine dei millisecondi anziché dei secondi.
- ✓ **Esempio:** se un terminale deve trasmettere dei dati (relativi ad una riga digitata), il relativo microprocessore "componè" il numero, trasmetta la riga e "riaggancia".

Commutazioni ibride

- ✓ **Connessione di pacchetto a divisione di tempo:** è un tipo di commutazione di pacchetto in cui ogni nodo scandisce le proprie linee di ingresso a *rotazione*:
 - ciascun pacchetto, che arriva con frequenza fissa, viene immediatamente ritrasmesso sulla linea di output corretta, non appena è stata letta l'intestazione.
 - Utilizzando pacchetti di dimensione fissa ed una rigida sincronizzazione non è necessario alcuno spazio sul buffer semplificando notevolmente la tecnologia dei nodi.
- ✓ La maggiore virtù della commutazione a divisione di tempo è costituita dalle prestazioni offerte (velocità >100Mbit/s) e dal basso costo.

Prospettive future

- ✓ Fra gli obiettivi primari della attuale ricerca nel campo delle telecomunicazioni vi è quello della riprogettazione della rete telefonica esistente per renderla adatta a fornire **servizi integrati**.
- ✓ La rete telefonica, l'infrastruttura *primaria* per le comunicazioni internazionali, è stata infatti progettata per la trasmissione analogica vocale
 - ↓
- ✓ si sta rivelando inadeguata per soddisfare le esigenze moderne di comunicazione (trasmissione dati, video, facsimile..), pur rimanendo la rete in assoluto più utilizzata.

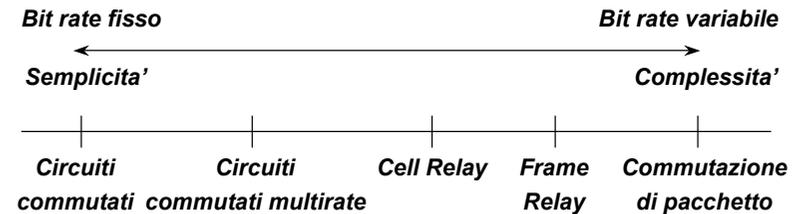
Prospettive future

- ✓ Sono nate, allora, l'idea e l'impegno internazionale per sostituire una grande porzione del sistema telefonico mondiale con un *sistema digitale avanzato*:
ISDN (Integrated Services Digital Network)
- ✓ Scopo dell' *ISDN* è l'integrazione di servizi vocali e di altro tipo su banda larga. I servizi offerti riguardano: il *videotelefono*, il *videotex*; il *facsimile*, la *telemetria* ..ecc.

Altre tecniche di commutazione

- ✓ Il *frame-relay* e il *cell-relay* sono due nuovi approcci alla trasmissione mediante tecniche di commutazione
- ✓ Nell'ambito delle tecniche di commutazione si configurano ad un livello intermedio, dove gli estremi rappresentano le tecniche tradizionali:
 - Commutazione di pacchetto
 - Commutazione di circuito
- ✓ Muovendoci verso sinistra abbiamo tecniche via-via meno complesse con bitrate fisso, mentre verso destra incontriamo tecniche caratterizzate da complessità crescente e bitrate variabile.

Altre tecniche di commutazione



Spettro delle tecniche di commutazione

Altre tecniche di commutazione

- ✓ Frame relay:
 - la procedura di controllo di flusso è stata resa più snella rispetto alle tecniche tradizionali a CP
- ✓ Cell relay:
 - pacchetti piccoli di dimensione fissa.
- ✓ Commutazione di circuito Multirate:
 - rappresenta una evoluzione della tecnica TDM.
 - la trasmissione si basa sulla commutazione di circuito e ha la capacità di instaurare connessioni mediante l'utilizzo di molteplici circuiti sincronizzati.
 - è una tecnica che fornisce maggiore flessibilità rispetto alla commutazione di circuito semplice ma ha lo svantaggio di presentare una notevole complessità.