**DATA LINK (Collegamento dati)**

Il suo compito è quello di ***organizzare*** il trasferimento dei dati tra ***due apparati adiacenti, logicamente connessi*** da un ***canale***, e di fornire una interfaccia definita per consentire allo strato di rete di ***accedere*** ai  ***servizi*** offerti. Si prende carico di diverse funzioni fra cui:

1. fornire un ben definito servizio d’interfaccia per lo strato network

2. gestire gli errori di trasmissione

3. regolare il flusso dati in modo che i dispositivi riceventi lenti non vengano sopraffatti dai trasmettitori veloci.

Per raggiungere questi obiettivi, lo strato data link prende i pacchetti provenienti dallo strato di rete e li incapsula in ***frame*** prima di trasmetterli. Ogni ***frame*** contiene un’intestazione (***header***), una sezione per contenere il pacchetto (***payload field***) e una sequenza di chiusura (***frame trailer***).

**Sorgente** **Destinatario**



La funzione del livello data link consiste nel fornire servizi allo strato di rete. Quello principale è quello di trasferire dati dallo strato network della macchina sorgente allo strato di rete della macchina destinazione.

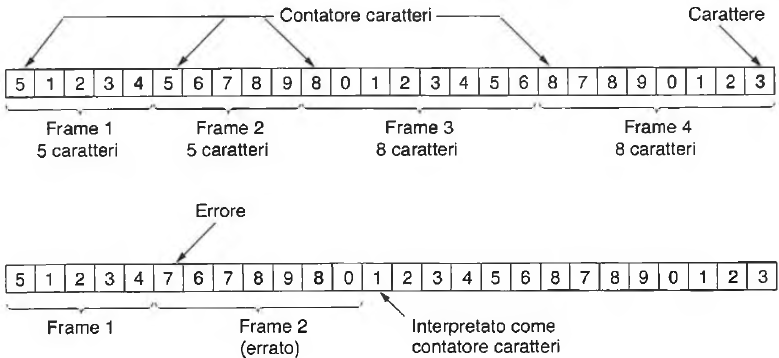
Tre servizi che vengono comunemente forniti a questo livello sono:

1. **servizio senza conferma *(unacknowledged)* senza connessione**, consiste nell’avere una macchina sorgente che invia dei frame indipendenti alla macchina destinazione, senza che quest’ultima debba dare conferma dell’avvenuta ricezione (es. Ethernet). L’uso di questa classe di servizio è legittimo quando la frequenza degli errori di trasmissione è molto bassa, così che la correzione può essere fatta dagli strati superiori.
2. **servizio con conferma** (***acknowledged***) **senza connessione,** questo genere di servizio continua a non usare nessun tipo di connessione logica, però ciascun frame è inviato individualmente e ne viene fatto *l’acknowledge* (*conferma della ricezione*). In questo modo il mittente riesce a sapere se un frame è arrivato a destinazione in modo corretto oppure no, con la possibilità di essere rispedito (es. reti wireless).
3. **servizio con conferma orientato alla connessione,** con questo tipo di servizio le macchine sorgente e destinazione stabiliscono una connessione prima d’iniziare a trasferire i dati. Ogni frame trasferito attraverso la connessione è numerato, e lo strato data link garantisce che venga effettivamente ricevuto. Risulta anche garantita la ricezione dei frame con l’ordine corretto. Quando si usa un servizio con connessione, il trasferimento dei dati avviene in tre fasi distinte. Nella prima fase viene stabilita la connessione; il mittente e il ricevente inizializzano variabili e contatori necessari per tenere traccia di quali frame sono stati ricevuti e quali no. Nella seconda fase uno o più frame vengono trasmessi. Nella terza fase la connessione viene rilasciata, liberando le variabili, i buffer e le altre risorse usate per mantenere la connessione (es. subnet di una WAN).

**Framing**

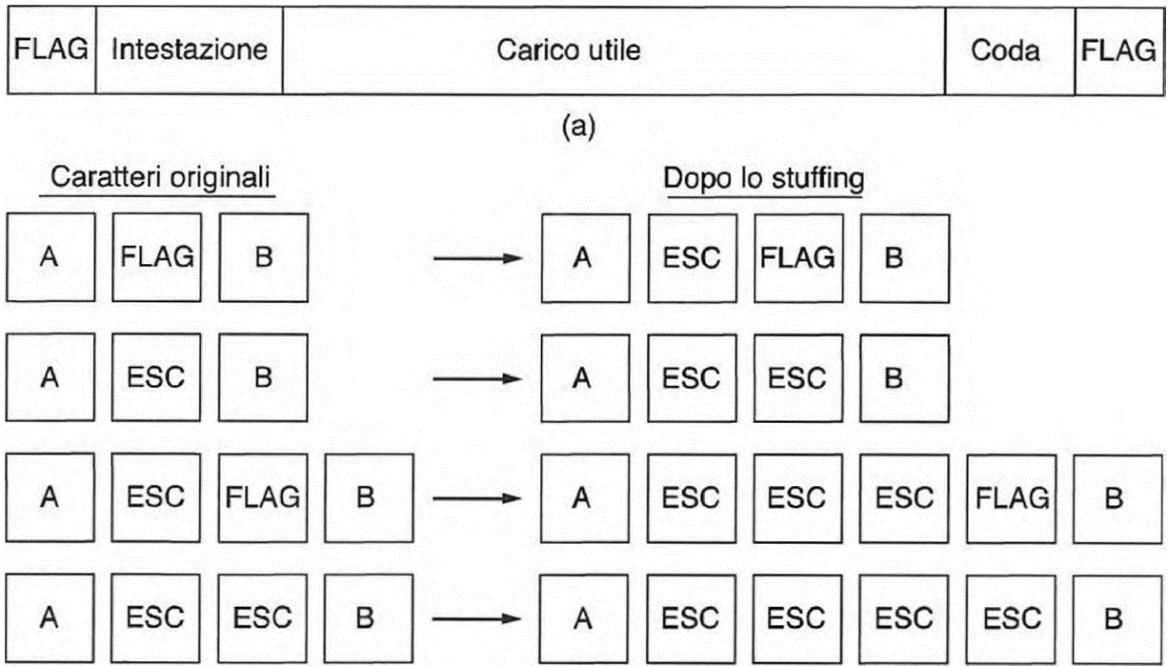
Il tipico approccio per lo strato data link è quello di suddividere il flusso di bit in una serie discreta di frame e calcolare il ***checksum*** per ogni frame. Esistono diverse tecniche:

1. ***conteggio dei caratteri***, usa un campo nell’intestazione per specificare il numero di caratteri nel frame. Quando lo strato data link della destinazione legge tale numero, sa quanti caratteri seguiranno e quindi sa dove si trova la fine del frame.



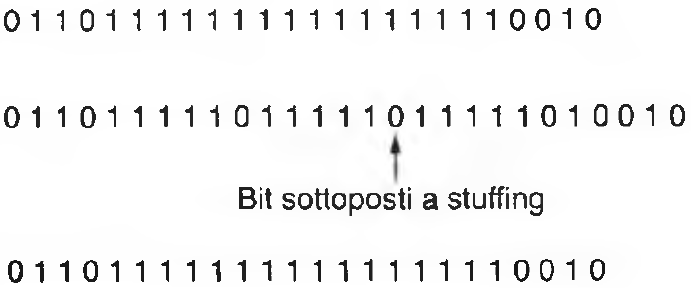
Il problema di questo algoritmo è che il conteggio può essere alterato da un errore di trasmissione.

1. ***flag byte con byte stuffing,*** introduce un byte speciale all’inizio e al termine di ogni frame chiamato ***flag byte***, per delimitare sia l’inizio sia la fine dei frame

******

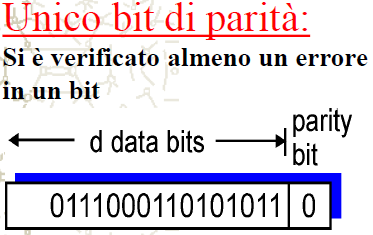
Può accadere che il valore corrispondente al flag byte compaia dentro ai dati, interferendo così con le operazioni di framing. La risoluzione consiste nel far sì che la sorgente inserisca un byte di ***escape*** (***ESC***) subito prima di ogni occorrenza del flag byte nei dati. Lo strato data link della destinazione provvederà a rimuovere i byte di escape prima di passare i dati allo strato network. Questa tecnica è chiamata ***byte stuffing*** o anche ***character stuffing***.

1. ***flag di inizio e fine con bit stuffing,*** ogni frame comincia e finisce con un gruppo speciale di bit, 01111110, che in sostanza è un flag byte. Ogni volta che lo strato data link della sorgente incontra cinque 1 consecutivi nei dati inserisce automaticamente un bit con valore 0 nel flusso in uscita.

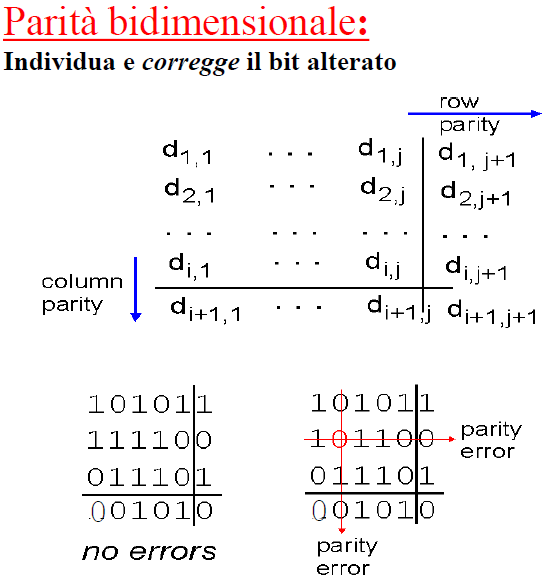


Questa operazione è chiamata ***bit stuffing.*** Quando la destinazione riceve cinque bit consecutivi con valore 1 seguiti da uno 0, automaticamente elimina lo 0*.*

**Controllo degli errori**

Il controllo dell’errore si basa su codici di ridondanza, che aggiungono bit alla parola dati per verificarne la correttezza. Tali codici si suddividono in:

* ***codici rilevatori***: in grado unicamente di rilevare la presenza o meno di errori nel frame, ma non la loro posizione, in questo caso il ricevente può chiedere la ritrasmissione del messaggio.



* ***codici correttori***: in grado di rilevare una o più posizioni errate nel frame e quindi di correggerle per semplice inversione del bit.

Vi sono 3 principali codici a rilevazione d'errore, diversi tra loro:

1. ***Bit di parità***, sono in grado di rilevare la presenza di un numero dispari di errori. Il bit di parità aggiunto assume il valore 0 o 1 per rendere pari il numero di “1” nella sequenza da trasmettere (parità pari) o per renderlo dispari (parità dispari). Per calcolare il bit di parità è sufficiente effettuare l’XOR dei bit di dato nella parola nel caso di parità pari e negare tale risultato nel caso di parità dispari. La parola trasmessa sarà formata dalla parola originale e dal bit di parità. Il ricevitore provvederà a ricalcolare la parità sulla configurazione ricevuta (escludendo il bit di parità aggiunto): confrontando il nuovo bit di parità con quello ricevuto è possibile stabilire se la trasmissione è avvenuta correttamente o no.
2. ***Check-sum***, si basa sull'aggiunta di simboli calcolati non sui singoli codici delle parole che costituiscono la comunicazione, ma valutati su un blocco di parole. L'obiettivo è quello di ottenere la più alta possibilità di rilevare errori con la minor ridondanza introdotta. La Check-sum è una delle tecniche di rilevazione errori maggiormente utilizzata per trasmissione a breve distanza. Dato un blocco di informazioni da codificare, la parola di controllo si calcola effettuando la somma in algebra modulo 2 (XOR) di tutti i codici delle singole parole. Il risultato è un byte che viene anch’esso trasmesso. Il ricevitore si preoccuperà di ripetere l’operazione sul pacchetto ricevuto confrontando il suo risultato con quello inviatogli dal trasmettitore: se le due somme coincidono significa che la trasmissione è avvenuta senza interferenze.
3. ***Codici di ridondanza*** ***ciclica (CRC) o codifica polinomiale***, È un altro metodo per la rilevazione degli errori è quello dei codici ciclici. Gli n bit del blocco da trasmettere vengono considerati come coefficienti di un polinomio di grado n-1 nella variabile x. Tale polinomio, che chiameremo M(x), viene poi diviso per un altro polinomio fissato dalle convenzioni internazionali, chiamato polinomio generatore e indicato con G(x). Per rilevare la presenza di un errore il ricevitore divide il messaggio ricevuto per G(x) e verifica che il resto sia nullo. Se non lo è il ricevitore deve chiedere la ripetizione del messaggio.

**Controllo di flusso 3.1.4**