

Università di Roma Tor Vergata  
Corso di Laurea triennale in Informatica  
**Sistemi operativi e reti**  
A.A. 2019-2020

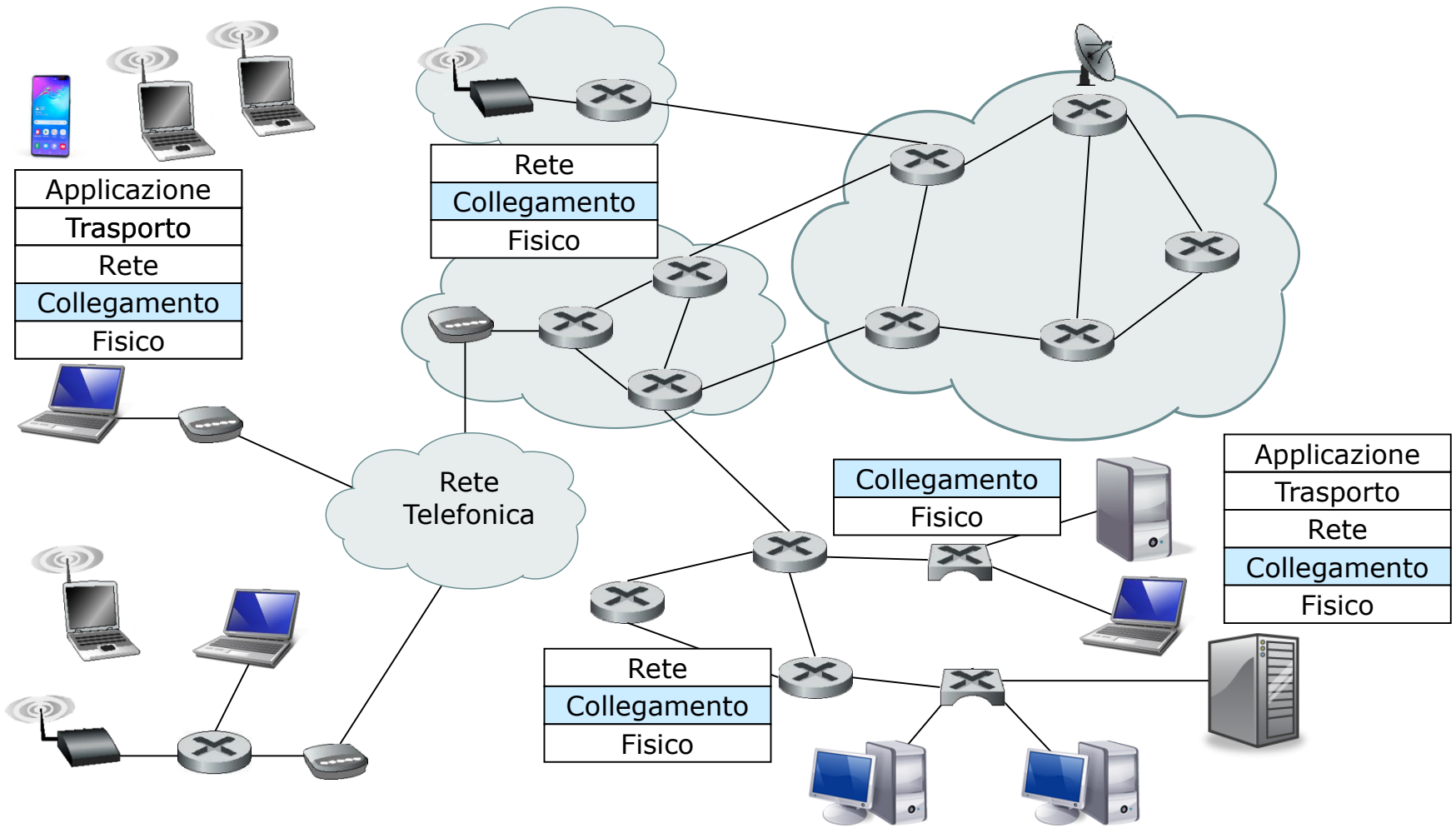
Pietro Frasca

**Parte II: Reti di calcolatori**  
**Lezione 21 (45)**

Giovedì 21-05-2020

# Lo strato di collegamento

- In questo strato indicheremo con il termine **nodi** sia gli host che i router.
- Le unità di dati (PDU-2) inviate da un protocollo dello strato di collegamento sono chiamate **frame (trame)**.
- Mentre lo strato di rete si occupa di trasferire segmenti dello strato di trasporto **da host a host**, un protocollo dello strato di collegamento esegue il trasferimento di datagram dello strato di rete **da nodo a nodo** per ogni **link** del percorso.
- Un nodo mittente incapsula il datagram in un **frame**, e lo trasmette su un link; un nodo destinatario riceve il frame ed estrae il datagram.
- Alcuni protocolli dello strato di collegamento molto diffusi sono: **Ethernet (IEEE 802.3)**, **LAN wireless 802.11**, **token ring e PPP**.



- Una caratteristica importante dello strato di collegamento è che un datagram può essere incapsulato **in diversi protocolli dello strato di collegamento** nei differenti link del percorso.
- Per esempio, un datagram può essere trasportato da PPP sul primo link, da wireless 802.11 (wi-fi) sull'ultimo link, e da Ethernet su tutti i link intermedi.
- È da notare che i vari protocolli dello strato di collegamento possono offrire servizi diversi allo strato soprastante. Per esempio, un protocollo dello strato di collegamento può essere affidabile nel trasferimento dati mentre un altro può non esserlo.

# I servizi forniti dallo strato di collegamento

- I principali servizi offerti da un protocollo dello strato di collegamento sono: il **trasferimento affidabile dei dati**, il **controllo di flusso**, la **verifica di errori**, e l'**accesso casuale al canale condiviso**.

## *Trasferimento affidabile*

- In modo analogo al TCP, l'affidabilità è ottenuta per mezzo di riscontri e ritrasmissioni.
- Un servizio per la trasmissione affidabile dei dati è generalmente usato per collegamenti che sono soggetti ad **alti tassi di errore**, come può essere un **link wireless**, con l'obiettivo di rilevare e/o correggere localmente, a livello di protocollo di collegamento, un errore piuttosto che propagarlo ai protocolli degli strati superiori e affidare quindi a un protocollo dello strato di trasporto o di applicazione il problema della rilevazione ed eventuale correzione.

- Generalmente, il servizio di trasmissione affidabile dello strato di collegamento non si utilizza nelle tecnologie per reti cablate (ad esempio Ethernet) nelle quali i collegamenti sono realizzati con mezzi trasmissivi guidati come ad esempio fibre ottiche e cavi UTP formati da doppiini in rame. Il servizio non è ritenuto necessario in quanto le reti cablate hanno bassi valori del tasso di errore.

## ***Accesso a link condivisi***

- Un **protocollo di controllo di accesso al mezzo (MAC, *Media Access Control Protocol*)** specifica le regole di trasmissione di un frame sul link. Per i collegamenti punto-punto che hanno un solo trasmettitore da un lato e un solo ricevitore dall'altro lato del link, il protocollo di accesso al link è semplice poiché il trasmettitore può inviare un frame ogni volta che il link non è occupato.
- Il caso più complesso si verifica quando più nodi condividono uno stesso link broadcast: il cosiddetto **problema dell'accesso multiplo.**

In questo caso il protocollo MAC serve a coordinare la trasmissione di frame da diversi nodi.

## ***Framing***

- Un frame è costituito da un **campo dati**, in cui è inserito il datagramma dello strato di rete, e da **campi di intestazione**. Il protocollo di collegamento specifica la struttura del frame. Le intestazioni del frame contengono anche campi per l'**indirizzo fisico** del nodo, che è *diverso* dall'indirizzo del nodo dello strato di rete (per esempio, IP).

## ***Controllo di flusso***

- I nodi in ciascun lato di un link hanno buffer di limitata capacità per contenere i frame. Questo può essere un problema, poiché un nodo ricevente, in certi momenti, potrebbe ricevere frame a una velocità che supera la sua capacità di elaborazione. Senza controllo di flusso, il buffer di ricezione potrebbe saturarsi e i frame essere scartati come visto per lo strato di trasporto.

## ***Rilevazione di errori***

- Un nodo ricevente può ricevere frame con errori nei bit. Questi errori sono generati dall'attenuazione del segnale e da disturbi elettromagnetici. Per non consegnare un datagramma che contiene errori, molti protocolli dello strato di collegamento implementano funzioni per rilevare la presenza di errori. Ciò si realizza con l'inserimento, nel nodo mittente, di un gruppo di bit per la rilevazione di errori, e con il nodo ricevente che compie un controllo degli errori. Ricordiamo che lo strato di trasporto (protocolli UDP e TCP) sia quello di rete (protocollo IPv4 ma non IPv6) forniscono, pur se in forma limitata, una rilevazione dell'errore (a livello software). **La rilevazione nello strato di collegamento è più sofisticata ed è implementata in hardware.**

## ***Correzione degli errori***

- un nodo ricevente può determinare la presenza di errori nel frame, ma anche correggerli.

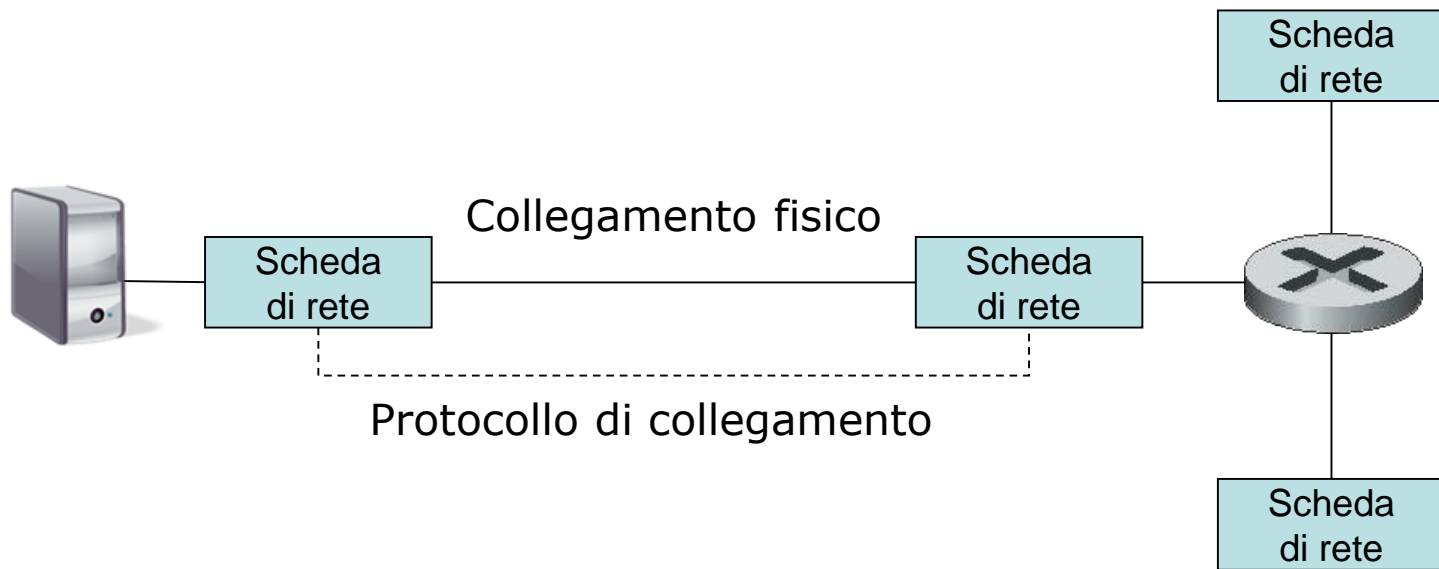


## ***Half-duplex e full-duplex***

- Con la trasmissione full-duplex i nodi di entrambe le estremità di un collegamento possono trasmettere frame contemporaneamente. Con la trasmissione half-duplex un nodo non può trasmettere e ricevere nello stesso tempo.

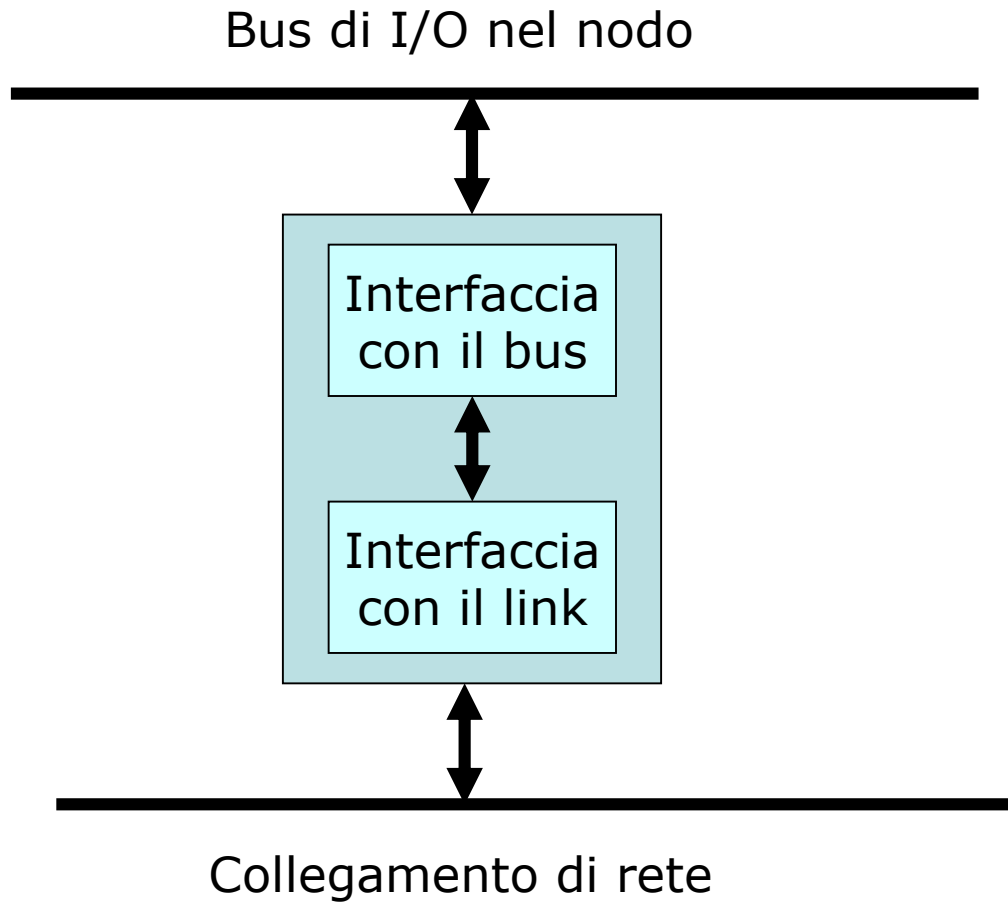
# Adattatori (schede di rete)

- Il protocollo dello strato di collegamento è, per la gran parte, implementato in un **adattatore** (*scheda di rete*).
- Un adattatore è un dispositivo elettronico composto da chip DSP (Digital Signal Processor), RAM, un'interfaccia per il bus dell'host (esempio PCI o USB) e un'interfaccia per il link.
- Come mostra la figura seguente, lo strato di rete nel nodo mittente passa un datagram all'adattatore il quale incapsula il datagram in un frame e lo trasmette nel link. Nel destinatario, l'adattatore riceve l'intero frame, estrae il datagram e lo passa allo strato di rete.
- Se il protocollo dello strato di collegamento implementa il controllo dell'errore, allora è l'adattatore mittente che esegue le operazioni per la rilevazione dell'errore ed è l'adattatore ricevente che esegue il controllo dell'errore.
- Se il protocollo dello strato di collegamento fornisce il trasferimento affidabile, allora il meccanismo (per esempio, numeri di sequenza, timer e riscontri) è implementato completamente dagli adattatori.



Il protocollo di collegamento è implementato nelle schede di rete

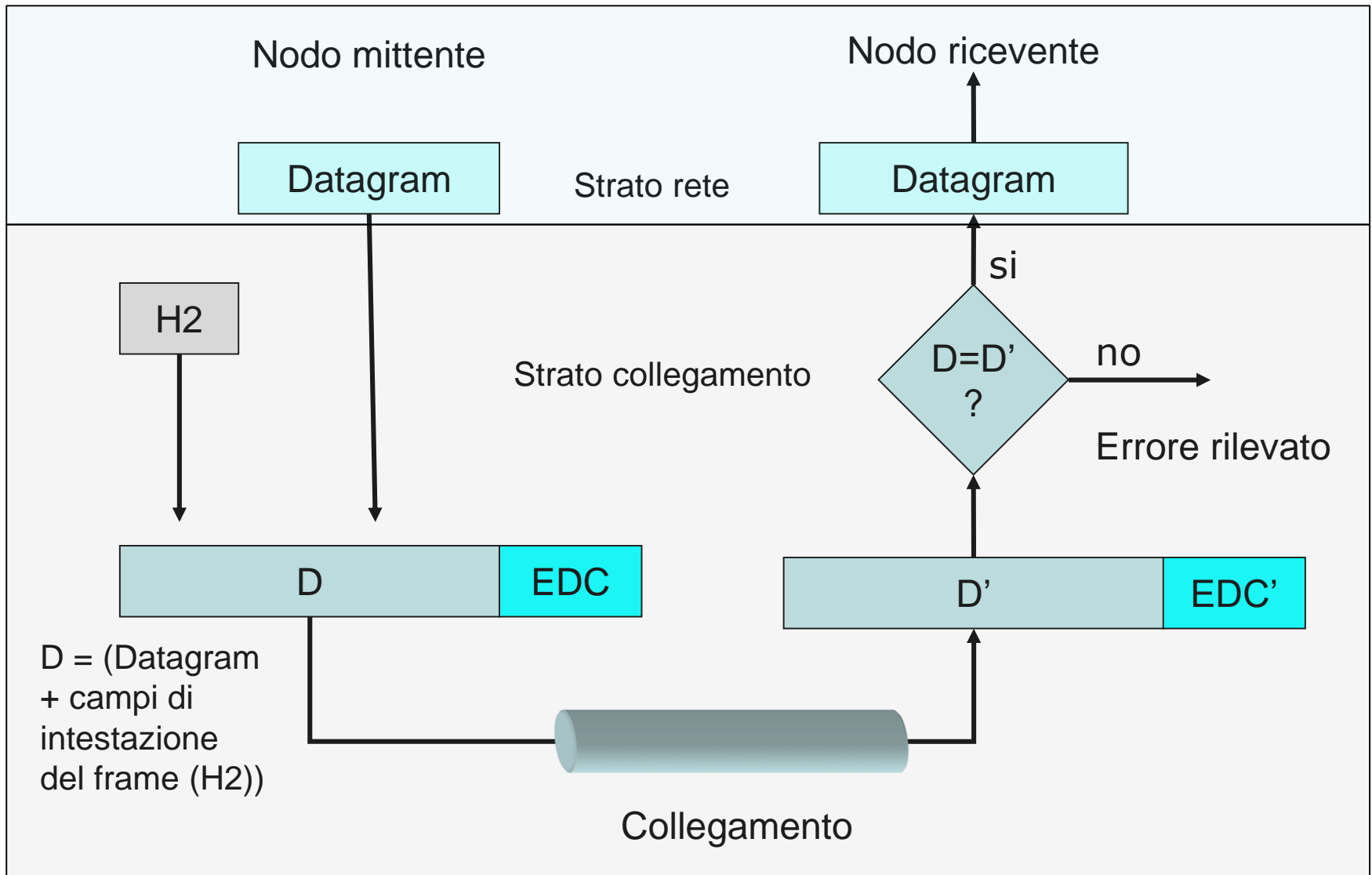
- Se il protocollo dello strato di collegamento fornisce l'accesso casuale il protocollo di accesso casuale è allora implementato completamente negli adattatori.
- Un adattatore è una **unità semi-autonoma**. In fase di trasmissione, un adattatore può ricevere un frame, determinare se ha errori e scartarlo senza passarlo allo strato soprastante. In modo analogo, in fase di trasmissione, un adattatore svolge il compito di trasmettere il datagram attraverso il collegamento cui è connesso.
- Comunque, un adattatore non è un'unità completamente autonoma. A livello hardware, esso è cablato sulla stessa main board (scheda madre) del dispositivo cui appartiene (o è connesso con esso tramite slot) ed è sotto il controllo del SO del dispositivo cui è connesso, mediante il suo driver.
- I componenti principali di un adattatore sono le interfacce con il bus e con il link. L'interfaccia con il bus è responsabile della comunicazione con il nodo collegato all'adattatore.



- L'interfaccia con il link è responsabile dell'implementazione del protocollo dello strato di collegamento. Oltre a framing e de-framing dei datagram, essa può fornire la rilevazione dell'errore, l'accesso casuale e altre funzioni dello strato di collegamento.
- La scheda **Ethernet** è la più diffusa tecnologia dello strato di collegamento.

# Tecniche di rilevazione e correzione degli errori

- La figura seguente rappresenta uno schema di base per lo studio delle tecniche di rilevazione e correzione degli errori.
- Nel nodo mittente, ai dati ***D*** da inviare, che devono essere protetti da possibili errori, sono aggiunti un certo numero di bit per la rilevazione e correzione dell'errore, ***EDC (Error Detection and Correction)***.
- Generalmente, i dati da proteggere sono i bit dell'intero frame, costituito dal datagramma dello strato di rete e dall'intestazione del protocollo dello strato di collegamento.
- ***D*** e ***EDC*** formano un frame che il nodo mittente invia al nodo ricevente.
- Il nodo ricevente, riceve una sequenza di bit ***D'*** ed ***EDC'*** che potrebbero essere diversi dagli originali ***D*** e ***EDC***, a causa di qualche **errore di trasmissione**.



## Rilevazione e correzione degli errori

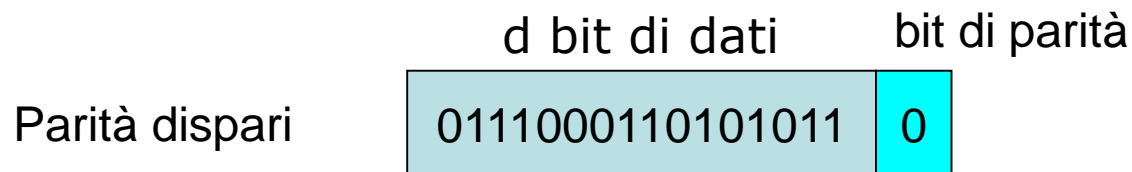
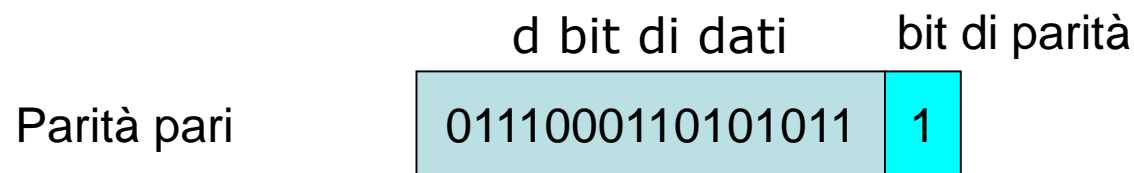


- Il compito del ricevente è di determinare se  $D'$  è uguale all'originale  $D$ , considerando il fatto che esso ha ricevuto  $D'$  e  $EDC'$ .
- Le tecniche di **rilevazione e di correzione degli errori** permettono al nodo ricevente di individuare se si sono verificati errori nei bit.
- Tuttavia, anche con l'uso dei bit EDC c'è la probabilità che ci siano errori non rilevati. Pertanto, il protocollo di collegamento nel lato ricevente potrebbe consegnare un datagramma errato allo strato di rete.
- In genere, per limitare la probabilità che non siano individuati errori nei bit, le tecniche di rilevazione e correzione degli errori più potenti e sofisticate utilizzano molti bit EDC.

- Esaminiamo ora tre tecniche per rilevare gli errori nei dati trasmessi:
  - **controllo di parità** (*parity check*) per spiegare i concetti di base della ricerca e della correzione dell'errore;
  - **metodi di checksum**, (metodi della somma di controllo) di solito usati nello strato di trasporto;
  - **controlli a ridondanza ciclica** (**CRC**, *cyclic redundancy check*) generalmente impiegati nello strato di collegamento negli adattatori di rete.

# Controllo di parità

- Una semplice tecnica di rilevamento degli errori usa un solo bit, detto **bit di parità**.
- Supponiamo che l'informazione da spedire ***D*** contenga ***d*** bit. In uno schema di **parità pari**, il nodo trasmittente aggiunge **un bit** di controllo e sceglie il suo valore in modo che il numero totale di numeri **1** nei ***d* + 1** bit (l'informazione originale più un bit di parità) **sia pari**. Analogamente, per lo schema di **parità dispari**, il valore del bit di parità è scelto in modo che ci siano un numero **dispari di 1**.



- Con un solo bit di parità l'operazione che esegue il ricevitore è semplice in quanto esso deve solo contare il numero di 1 nei bit  $d + 1$  ricevuti. Nel caso di schema di parità pari, il ricevitore se trova un numero dispari di bit con valore 1 può stabilire che si è verificato almeno un errore in un bit. Più esattamente, può stabilire che si è verificato **un numero dispari di errori**.
- Se si verifica un **numero pari di errori** il ricevitore non è in grado di rilevare l'errore.
- Se il canale fisico è molto affidabile e quindi la probabilità di errore è piccola e si può assumere che un errore possa verificarsi in un bit indipendentemente dal successivo, la probabilità di errori multipli in un frame sarà molto bassa. In questo caso, un solo bit di parità potrebbe essere sufficiente.
- Però, le statistiche sulle misure hanno dimostrato che spesso gli errori si presentano a "**raffica**", invece che verificarsi indipendentemente tra loro.

- Se gli errori si verificano a raffica, la probabilità di non rilevare gli errori in un frame protetto da un solo bit di parità è del **50%**, in quanto è in ugual misura la probabilità che il numero di bit alterati sia pari o dispari.
- Per tali situazioni, è necessario ricorrere ad una tecnica più potente e sofisticata per la rilevazione degli errori.
- Consideriamo ora un'estensione dello schema di parità a un bit per mostrare una semplice soluzione della tecnica di **correzione dell'errore**.
- La figura seguente mostra come, i  **$d$**  bit in  **$D$**  sono suddivisi in  **$i$  righe** e  **$j$  colonne**. Un valore di parità è calcolato per ciascuna riga e per ciascuna colonna. I risultanti  **$i + j + 1$**  bit di parità contengono i bit per la rilevazione dell'errore del frame del link dati.

$d_{1,1}$	$d_{1,2}$	.....	$d_{1,j}$	$d_{1,j+1}$
$d_{2,1}$	$d_{2,2}$	.....	$d_{2,j}$	$d_{2,j+1}$
.....				
.....				
$d_{i,1}$	$d_{i,2}$	.....	$d_{i,j}$	$d_{i,j+1}$
$d_{i+1,1}$	$d_{i+1,2}$		$d_{i+1,j}$	$d_{i+1,j+1}$

$D = 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1$

Senza errori

1	0	1	0	0
1	1	0	1	1
1	0	0	1	0
0	1	0	1	0
1	0	1	1	1

Errore correggibile (1 bit)

1	0	1	0	0
1	1	0	1	1
<del>1</del>	<del>1</del>	0	1	0
0	1	0	1	0
1	0	1	1	1

Parità pari a due dimensioni.

- Supponiamo ora che si verifichi un solo errore nei bit degli originali ***d*** bit dei dati. Con questo schema di parità a due dimensioni, le parità sia della colonna sia della riga che contengono il bit errato presenteranno l'errore.
- Il ricevitore può quindi non solo ***rilevare*** che si è verificato un errore in un bit, ma può usare gli indici della colonna e della riga che presentano l'errore di parità per identificare il bit che è stato modificato e ***correggerlo***.
- La figura precedente mostra un esempio in cui il bit di valore **0** in posizione **(3,2)** ha cambiato valore passando da 0 a 1.
- L'errore può essere rilevato e corretto anche se si verifica nei bit di parità.
- La parità a due dimensioni può anche rilevare, ma non correggere, qualsiasi combinazione di due errori in un pacchetto.
- Per la loro semplicità, le tecniche di controllo di parità sono principalmente usate nei collegamenti altamente affidabili, come ad esempio nei bus SCSI e PCI dei computer e in molte cache di microprocessori.

- Tuttavia, queste tecniche sono anche usate nelle trasmissioni seriali dei dati (comunicazioni tra terminali e server), dove viene usato comunemente un formato di 7 bit, con un bit di parità pari e uno o due bit di stop. Inoltre, il controllo di parità è anche usato in dispositivi audio di registrazione e riproduzione, come nei CD audio.
- La capacità del ricevitore di **rilevare e correggere** gli errori è detta **FEC** (**FEC**, *Forward Error Correction*, correzione degli errori in avanti).
- Le tecniche FEC sono fondamentali perché possono diminuire il numero di ritrasmissioni richieste al nodo trasmittente. Infatti con la correzione automatica dell'errore il ricevitore non deve inviare indietro al mittente un frame **NAK** (negative acknowledgment) e quindi attendere che il mittente ritrasmetta il frame errato, ritrasmissione che invece è necessaria nel caso di sola rilevazione dell'errore.



## Metodi di checksum

- Le tecniche basate sulla checksum suddividono i pacchetti di  $d$  bit da proteggere in parole di  $k$  bit.
- Come già descritto per UDP e TCP, la *checksum di Internet* si basa su questa tecnica: il nodo mittente calcola la checksum facendo il complemento a 1 della somma delle parole di 16 bit del segmento e di alcuni campi dell'intestazione IP.
- Il nodo ricevente controlla la checksum effettuando il complemento a 1 della somma dei dati ricevuti (compresa la checksum) e verifica che il risultato sia formato da tutti bit pari a 1. Se qualche bit è pari a 0, viene indicato un errore.
- I metodi checksum forniscono una protezione debole contro gli errori in confronto alle tecniche CRC, che generalmente sono usate nei protocolli dello strato di collegamento.

- Il motivo per cui il metodo checksum è implementato nello strato di trasporto è perché questo è implementato a livello software e fa parte del sistema operativo e quindi è necessario utilizzare una tecnica di rilevazione di errore semplice e veloce.
- La rilevazione dell'errore nello strato di collegamento è invece **implementato in hardware nelle schede di rete**, per cui si possono effettuare più rapidamente le più complesse operazioni di controllo di errori.
- Inoltre è meglio correggere l'errore immediatamente nell'adattatore piuttosto che propagarlo al sistema operativo.

# Controllo a ridondanza ciclica (CRC)

- Una tecnica di rilevazione dell'errore molto usata nelle reti di calcolatori è basata sui codici di controllo a ridondanza ciclica (**CRC, Cyclic Redundancy Check**).
- I codici CRC sono chiamati anche **codici polinomiali**, poiché è possibile vedere la stringa dei bit da controllare come un polinomio i cui coefficienti hanno valori 0 e 1.
- Ad esempio, la sequenza **10101101** può essere rappresentata dal polinomio

$$x^7 + x^5 + x^3 + x^2 + 1$$

- Per descrivere il funzionamento dei codici CRC consideriamo una sequenza ***D***, costituita di ***d*** bit che il nodo mittente invia al nodo ricevente.



- Il mittente appende alla stringa ***D*** di lunghezza ***d*** bit, una stringa ***R*** di ***r*** bit, in modo che la stringa risultante di ***d + r*** bit sia esattamente divisibile per un polinomio ***G*** di ***r+1*** bit detto ***polinomio generatore***.
- Trasmettitore e ricevitore utilizzano uno stesso polinomio generatore ***G*** di grado ***r+1***, ed eseguono le operazioni con l'aritmetica modulo 2.



- Il ricevitore divide i  $d + r$  bit ricevuti per  $G$ .
- Se il resto della divisione è diverso da zero, il ricevitore capisce che si è verificato un errore; altrimenti i dati sono considerati corretti.
- Tutti i calcoli CRC sono eseguiti in **aritmetica modulo 2** senza riporti e resti in addizione e sottrazione.



- Questo significa che addizione e sottrazione sono operazioni identiche, e che entrambe equivalgono all'**operazione di OR esclusivo (XOR)** bit a bit degli operandi.

x1	x2	Y
0	0	0
0	1	1
1	0	1
1	1	0

- Allora, per esempio,

$$1011 \text{ XOR } 0101 = 1110$$

$$1001 \text{ XOR } 1101 = 0100$$

- Moltiplicazione e divisione sono le stesse dell'aritmetica in base 2. La moltiplicazione per  $2^r$  provoca lo slittamento (shift) a sinistra dei bit di  $r$  posizioni. Quindi, dati  $D$  e  $R$ , la quantità  $D \cdot 2^r \text{ XOR } R$  fornisce una stringa di  $d + r$  bit.



$D \cdot 2^r \text{ XOR } R$

- Il **mittente** calcola  $R$  in modo che  $D \cdot 2^r \text{ XOR } R$  sia divisibile per  $G$  senza avere resto:

$$(D \cdot 2^r \text{ XOR } R) / G = n =$$

$$D \cdot 2^r \text{ XOR } R = n \cdot G$$

Se eseguiamo l'operazione XOR di **R** con entrambi i membri dell'equazione sopra, otteniamo

$$D \cdot 2^r \text{ XOR } R = n \cdot G$$

$$\begin{aligned} D \cdot 2^r \text{ XOR } R \text{ XOR } R &= n \cdot G \text{ XOR } R = \\ D \cdot 2^r \text{ XOR } (R \text{ XOR } R) &= n \cdot G \text{ XOR } R = \\ D \cdot 2^r \text{ XOR } 0 &= n \cdot G \text{ XOR } R = \\ D \cdot 2^r &= n \cdot G \text{ XOR } R \end{aligned}$$

- Questa equazione ci dice che se dividiamo  $D \cdot 2^r$  per **G**, il valore del resto è precisamente **R**.
- In altre parole, possiamo calcolare **R** come

$$R = \text{resto } (D \cdot 2^r / G)$$



# Esempio di calcolo CRC

Eseguiamo il calcolo del CRC nel caso in cui

$$D = 101110 \text{ (} d = 6\text{)}$$

$$G = 1001, r = 3.$$

I nove bit trasmessi in questo caso sono **101110 011**.

**D·2<sup>3</sup>**

1 0 1 1 1 0	0 0 0
1 0 0 1	
<hr/>	
0 0 1 0 1 0	
1 0 0 1	
<hr/>	
0 0 1 1 0 0	
1 0 0 1	
<hr/>	
0 1 0 1 0	
1 0 0 1	
<hr/>	
0 0 1 1	

**R**

**G**

1 0 0 1	
1 0 1 0 1 1	
<hr/>	

1 0 1 1 1 0 0 1 1

Esempio di calcolo di CRC al mittente

<b>D'</b>		<b>G</b>
1 0 1 1 1 0 0 1 1		1 0 0 1
1 0 0 1		1 0 1 0 1 1
0 0 1 0 1 0		
1 0 0 1		
0 0 1 1 0 1		
1 0 0 1		
0 1 0 0 1		
1 0 0 1		
0 0 0 0		
<b>R</b>		

Esempio di calcolo di CRC al ricevitore

- Sono stati definiti standard internazionali per **generatori G** di 8, 12, 16 e 32 bit. Lo standard CRC-32 a 32 bit, che è stato adottato in numerosi protocolli IEEE dello strato di collegamento, usa un generatore di

$$\mathbf{G_{CRC-32} = 100000100110000010001110110110111}$$

- Ciascuno dei CRC standard può rilevare errori a raffica inferiori a  $r + 1$  bit.
- Inoltre, ciascuno dei CRC standard può rilevare qualsiasi numero dispari di errori.