



SAPIENZA  
UNIVERSITÀ DI ROMA



# Reti per l'automazione

Automazione

20/10/2015

Alessandro De Luca

# Il sistema di comunicazione

---

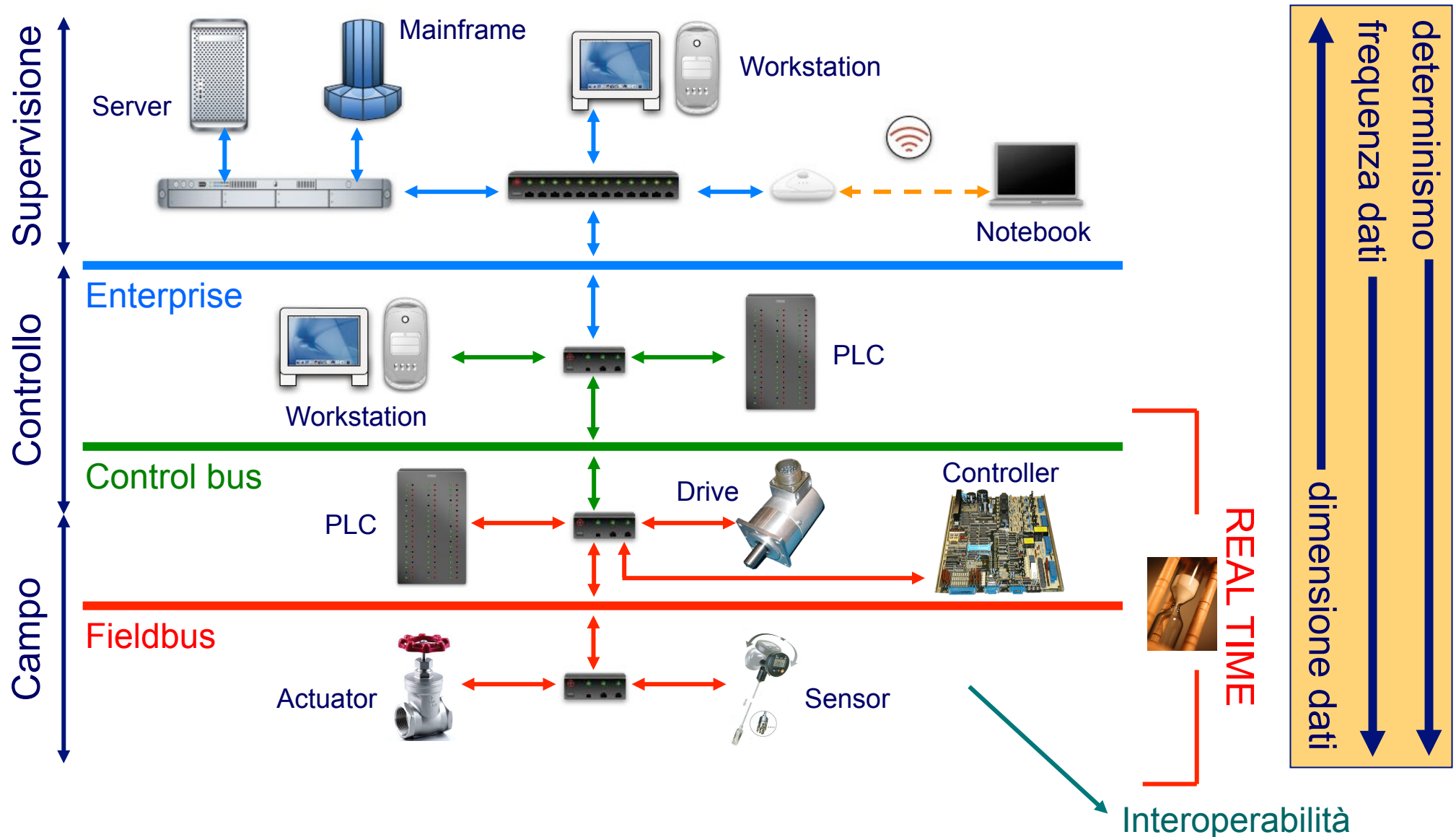
Ad ogni livello della piramide dell'automazione

- si acquisiscono informazioni
- si elaborano strategie
- si attuano azioni correttive

Importanza fondamentale del sistema di comunicazione

- l'intero sistema deve essere interconnesso per garantire il flusso di informazioni
  - ➔ comunicazione orizzontale
  - ➔ comunicazione verticale: gerarchia di RETI
  - ➔ caratteristiche diverse ai vari livelli
    - tipologia dei dati
    - vincoli di comunicazione

# Reti di comunicazione



## Rete Enterprise

- ❑ rete per le informazioni gestionali
- ❑ client (workstation, notebook, mainframe) e server sono standard
- ❑ non è real-time
- ❑ la sicurezza dell'informazione è importante, ma non la “robustezza” rispetto a disturbi ambientali (in genere, minimi)
- ❑ standard Ethernet
  - ➔ Ethernet è una tecnologia “connection-less”: non viene garantita la ricezione dei dati
  - ➔ è il protocollo TCP/IP che garantisce la ri-trasmissione dei dati nel caso di mancanza del segnale di acknowledgement relativo al singolo pacchetto

## Rete di Controllo e Rete di Campo

- ❑ rete per le informazioni di cella, macchina e campo
- ❑ il client non è standard (PLC, controllori embedded, dispositivi di campo) perché è importante poter avere flessibilità
- ❑ dati piccoli, non strutturati, ma trasmessi con frequenza elevata
- ❑ vincoli Real Time
  - ➔ Ethernet non va bene
  - ➔ necessità di soluzioni ad hoc
- ❑ necessità di determinismo
  - ➔ **ritardo di trasmissione**: introduce un ritardo negli anelli di controllo che portano ad un degrado delle prestazioni
- ❑ l'ambiente industriale "ostile" necessita robustezza
- ❑ impatto sia sulle caratteristiche HW (topologia) che su quelle SW (protocolli e servizi) delle reti

## Topologia

- ❑ **broadcast:** unico canale di comunicazione condiviso da tutte le macchine della rete
  - ➔ i pacchetti sono inviati dal trasmettitore sul canale e ricevuti da tutti gli altri host (con allocazione statica o dinamica del canale)
  - ➔ ciascun pacchetto contiene l'indirizzo del destinatario: ogni macchina controlla l'indirizzo e decide se il pacchetto è indirizzato a lei o può essere scartato
  - ➔ si possono indirizzare i pacchetti a tutte le macchine contemporaneamente (broadcasting) o anche solo a un sottoinsieme (multicasting)
- ❑ **punto-punto (peer-to-peer):** connessioni dedicate tra coppie di dispositivi
  - ➔ è necessario definire il cammino per trasmettere tra due macchine non fisicamente connesse

# Classificazione delle reti

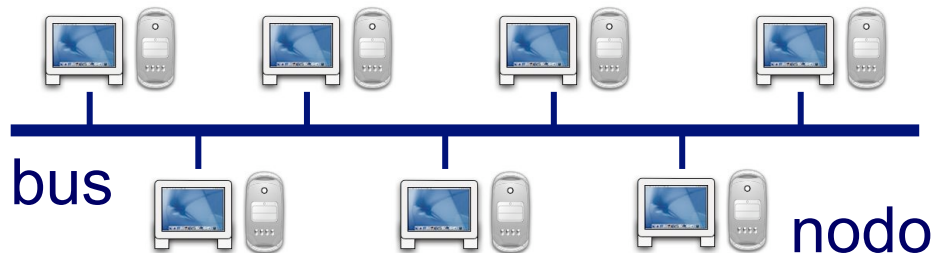
## Reti broadcast (tipicamente LAN)

### □ reti a bus

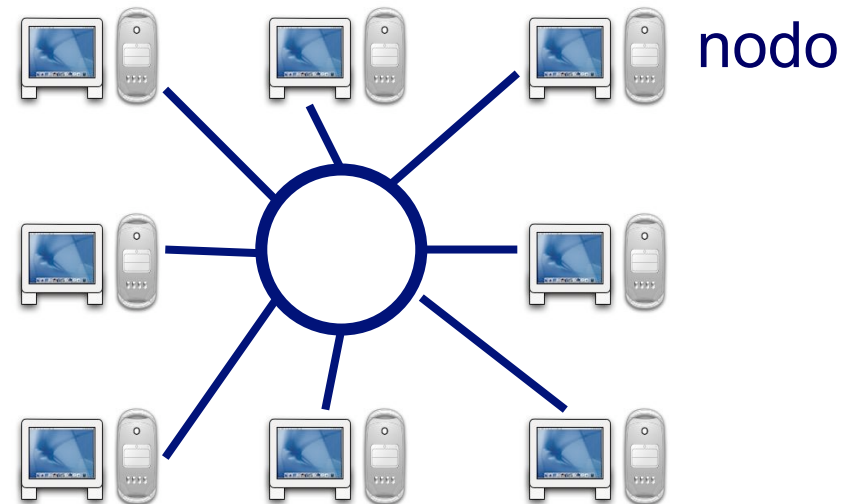
- ➔ in ogni istante un solo nodo dovrebbe trasmettere (master)
- ➔ necessario quindi arbitraggio, centralizzato o distribuito (più comune)

### □ reti ad anello

- ➔ i pacchetti circolano in serie sull'anello (ciascun bit in un tempo tipicamente inferiore al tempo di trasmissione dell'intero pacchetto)
- ➔ necessario arbitraggio per accessi simultanei all'anello



rete a bus (IEEE 802.3 = ethernet)



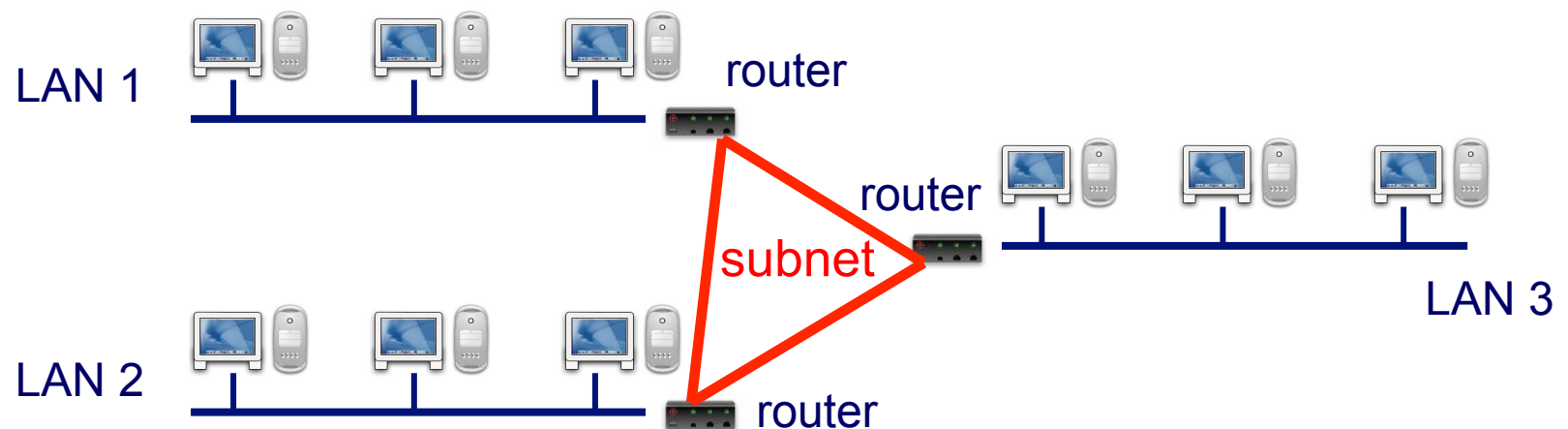
rete ad anello

(nodi con cablaggio locale per potersi disconnettere)

# Classificazione delle reti

## Reti peer-to-peer

- ❑ per reti di dimensioni maggiori, normalmente più reti locali (di tipo LAN) vengono connesse tramite una sottorete (subnet) di tipo punto-punto
  - ➔ ogni nodo della sottorete funge da router per la propria rete locale
  - ➔ pacchetti in ingresso sono memorizzati e poi inviati verso la destinazione finale
  - ➔ la problematica maggiore è quella della definizione del cammino che ogni pacchetto deve seguire per ottimizzare le prestazioni della rete (routing)
  - ➔ se più pacchetti relativi allo stesso messaggio seguono cammini diversi, è necessario gestire anche la sequenza con cui essi vengono ricevuti

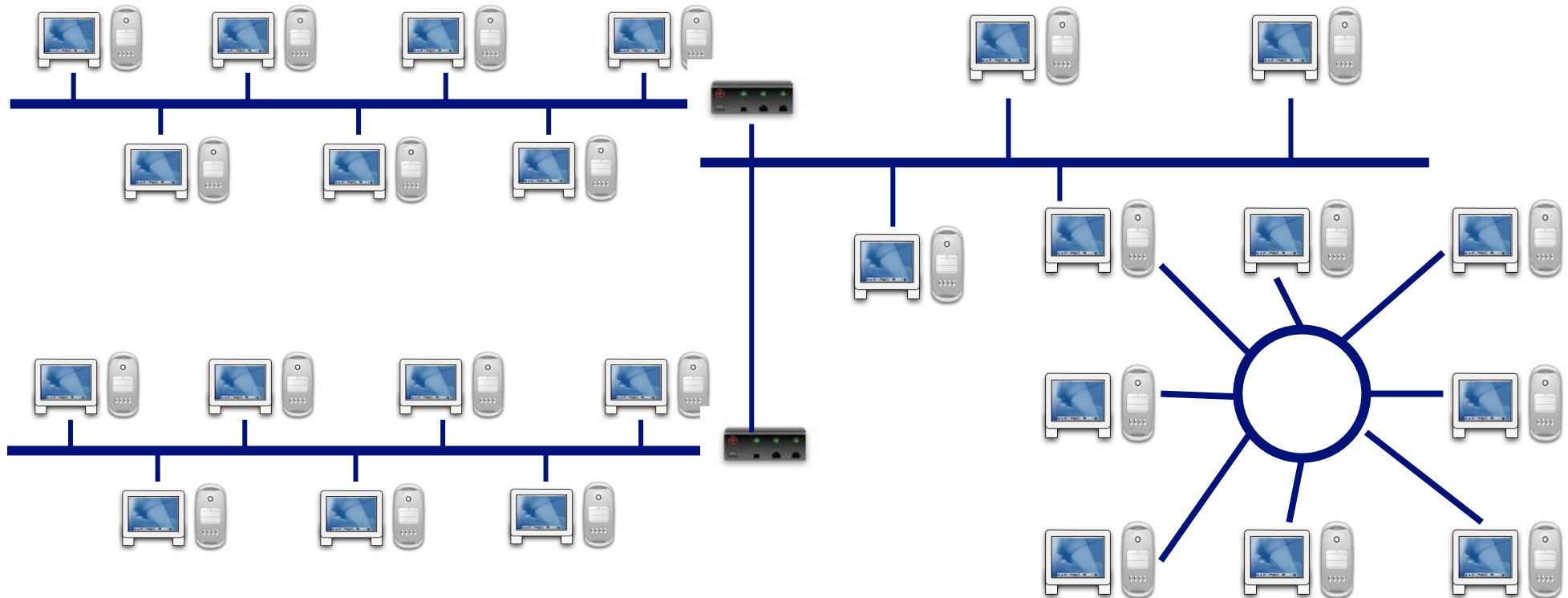




# Classificazione delle reti

## Reti ibride

- ❑ reti broadcast, anche diverse tra di loro, connesse tramite una rete ancora broadcast
  - ➔ è la situazione più frequente in applicazioni di controllo complesse



# Classificazione delle reti

## Estensione geografica

distanza media tra due nodi	estensione geografica
1 m	metro quadro
10 m	stanza
100 m	edificio
1 km	distretto
10 km	città
100 km	nazione
1,000 km	continente
10,000 km	pianeta

**Personal Area Network (PAN)**  
ad es., bluetooth

**Local Area Network (LAN)**

**Metropolitan Area Network**  
ad es., reti telefoniche o cable-TV **(MAN)**

**Wide Area Network (WAN)**

**Internet**

## Estensione geografica

### ❑ Local Area Network (LAN)

- ➔ reti private di dimensioni ridotte, normalmente all'interno di una istituzione e di dimensioni da pochi metri fino a pochi chilometri
- ➔ normalmente reti broadcast, con tempi e ritardi di trasmissione noti e limitati

### ❑ Wide Area Network (WAN)

- ➔ reti che coprono una area geografica estesa
- ➔ la connessione su larga scala è realizzata tramite sottoreti (subnet/provider)
- ➔ linee di trasmissione e router per la commutazione

### ❑ altri tipi di reti

- ➔ reti metropolitane, che coprono le dimensioni tipiche di una città
- ➔ internet di dimensione mondiale

### ❑ la dimensione di una rete ne limita le caratteristiche realizzative: reti estese **non** possono essere di tipo broadcast

### ❑ le reti di interesse per i sistemi di controllo sono normalmente LAN

## Protocollo di accesso al mezzo per reti **broadcast**

### ❑ **allocazione statica del canale**

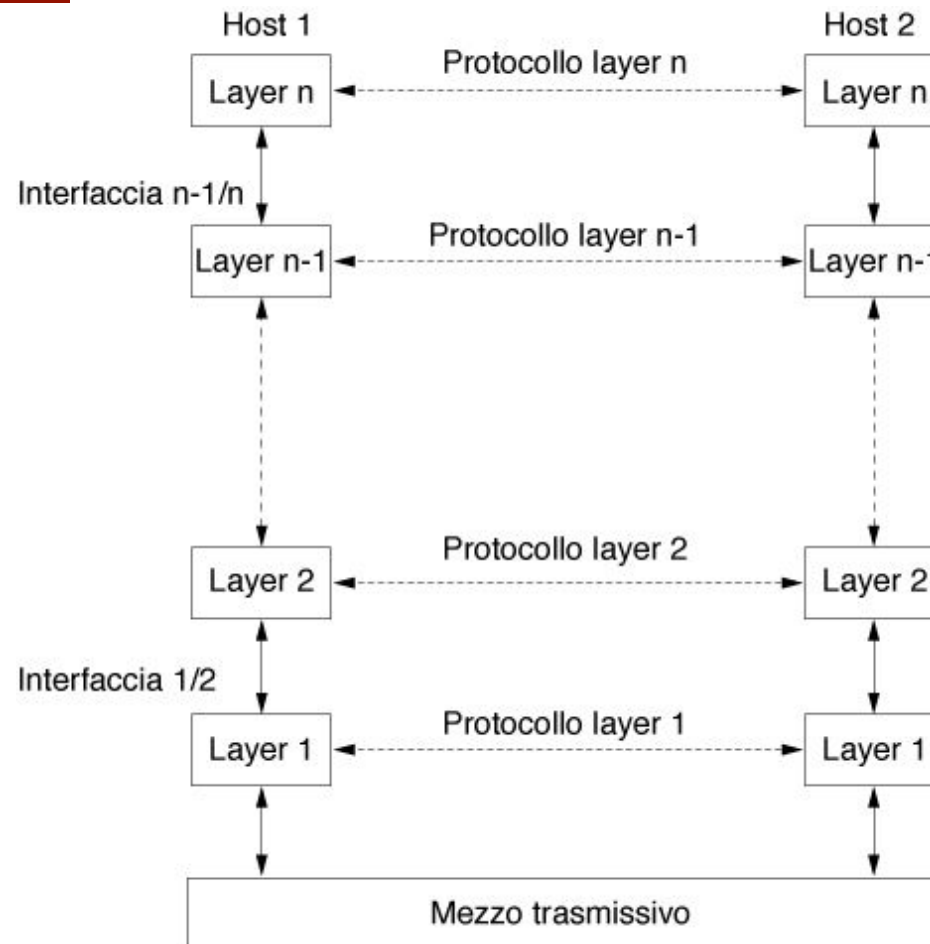
- ➔ il tempo viene suddiviso in “quanti” (time units), ed ogni nodo può eseguire il broadcast solamente in corrispondenza del quanto assegnato al nodo stesso
- ➔ se un nodo non ha nulla da trasmettere il quanto rimane inutilizzato:  
l’allocazione statica in questo caso non utilizza tutta la banda disponibile

### ❑ **allocazione dinamica del canale**

- ➔ controllo centralizzato: un master determina il prossimo nodo che trasmette
- ➔ controllo decentralizzato: ogni nodo decide autonomamente se iniziare a trasmettere
- ➔ sistemi **a collisione**: è possibile che più nodi trasmettano in modo contemporaneo (le collisioni devono essere rilevate e risolte)

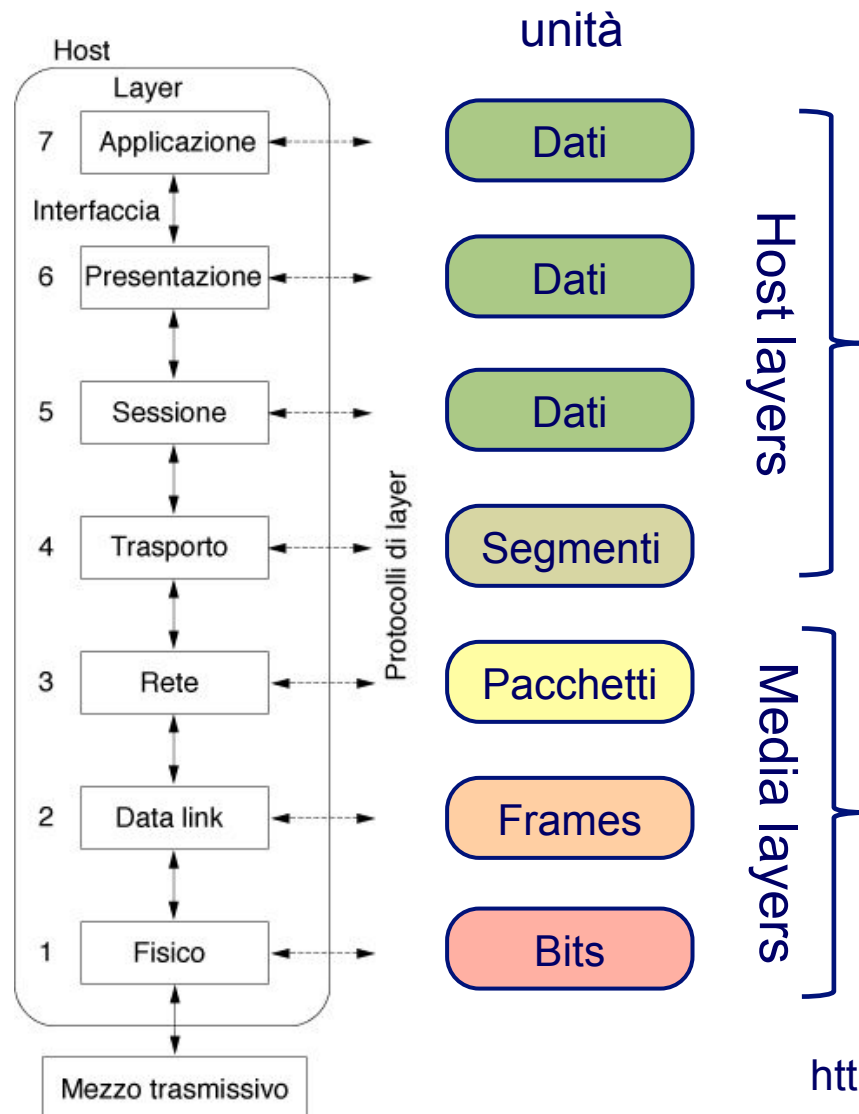
Organizzata a stack

- ❑ ad ogni livello corrisponde una funzionalità: layer
- ❑ il layer n di un host parla solo con il layer n di un altro host mediante una serie di regole dette protocolli



# Architettura ISO-OSI

International Organization for Standardization (ISO) - Open Systems Interconnection

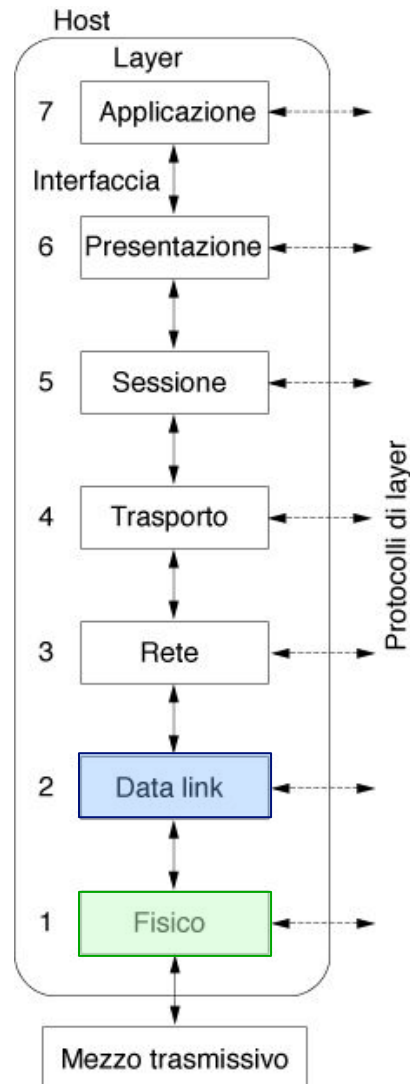


- ❑ dal 1983
- ❑ stack con 7 layer

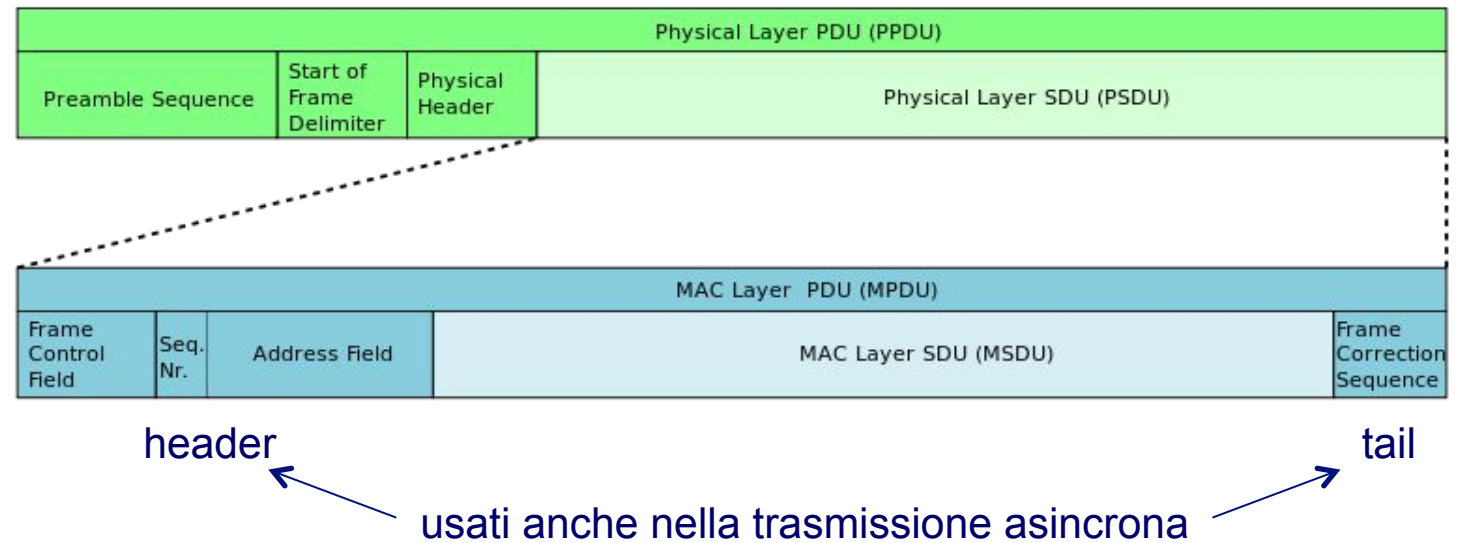


[http://it.wikipedia.org/wiki/Open\\_Systems\\_Interconnection](http://it.wikipedia.org/wiki/Open_Systems_Interconnection)

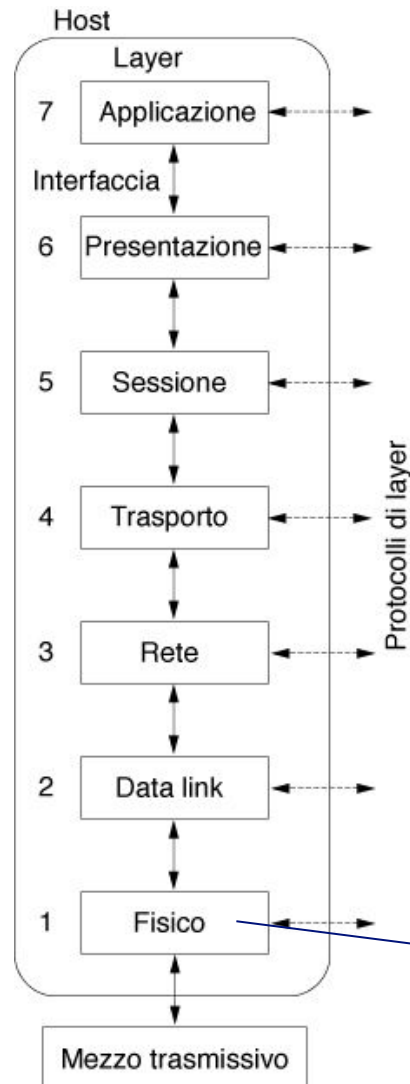
# Architettura ISO-OSI



- ogni livello comunica con il livello corrispondente dei nodi di transito o destinatari usando il *Service Access Point* (SAP) del livello sottostante
- messaggi del livello  $n$  sono “incapsulati” nei messaggi di livello inferiore  $n-1$  (fino a quello fisico)
- Service Data Units (SDU) (contenuto del messaggio) dentro Protocol Data Units (PDU)



# Architettura ISO-OSI



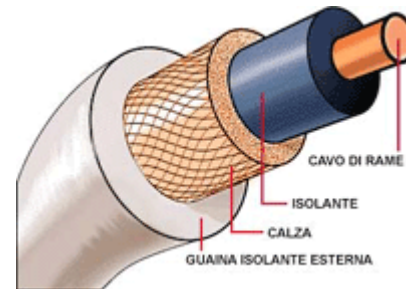
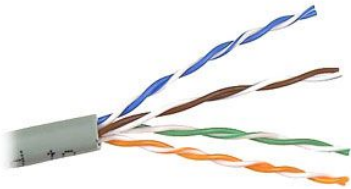
- definisce la relazione tra il dispositivo ed il mezzo di trasmissione (doppino di rame o fibra ottica)
  - layout dei pins, livelli di tensione, impedenza di linea, specifiche su cavi e connettori, codifica e temporizzazione dei segnali e loro modulazione e/o conversione A/D, ripetitori e adattatori di rete o di bus, ...
  - trasmissione sincrona (clock che determina anche la velocità di trasmissione) o asincrona (con bit di start e stop)
  - direzionalità: simplex (mono-), duplex (bi-), half-duplex (mono- , ma alternata)
- usato dal layer fisico di Ethernet (RJ45) e di altre LAN (token ring, IEEE 802.11) o PAN (Bluetooth)

**physical layer:** trasmette fisicamente i bit sul canale; si fissano le convenzioni fisiche ed elettriche



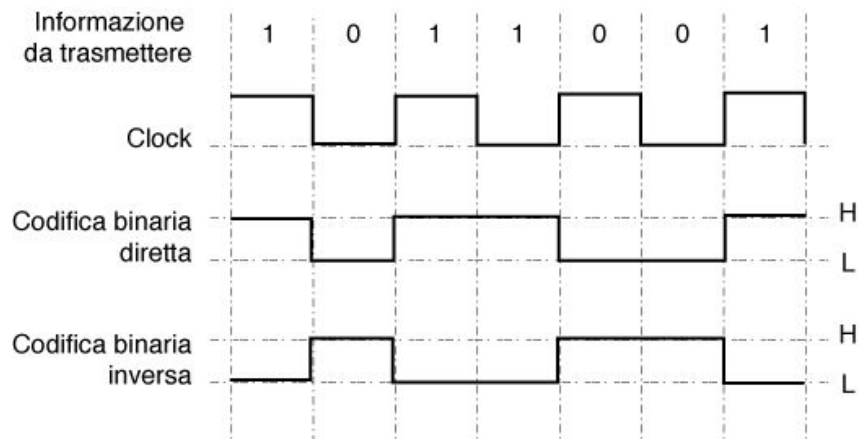
## Mezzi fisici per la trasmissione dei segnali

- ❑ **doppino telefonico: coppia di cavi di rame 'twisted'**
  - ➔ cavo 10 Base-T, 100 Base-T, ...
  - ➔ throughput (quantità di dati trasmessi in un lasso di tempo): Kbps
- ❑ **cavo coassiale: conduttore in rame circondato da materiale isolante**
  - ➔ buona banda e reiezione ai disturbi
  - ➔ throughput: Mbps
- ❑ **fibra ottica: l'informazione è tradotta in impulsi luminosi**
  - ➔ larghezza di banda decisamente superiore
  - ➔ throughput: Tbps
- ❑ **wireless**

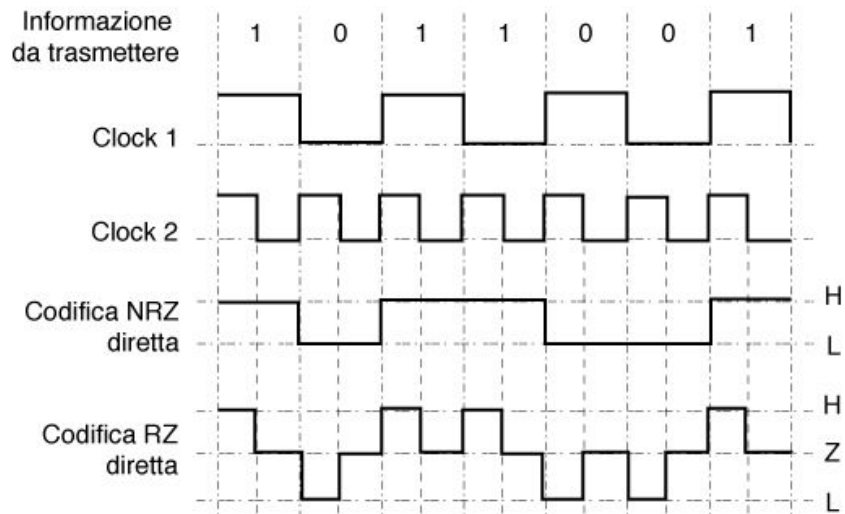


# Livello fisico della rete

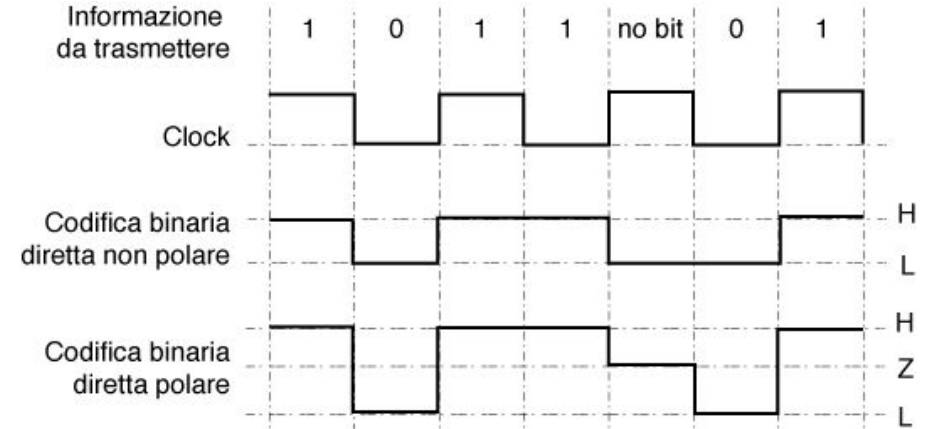
codifica dell'informazione logica dei bit (diversi spettri di frequenza, a media nulla o meno)



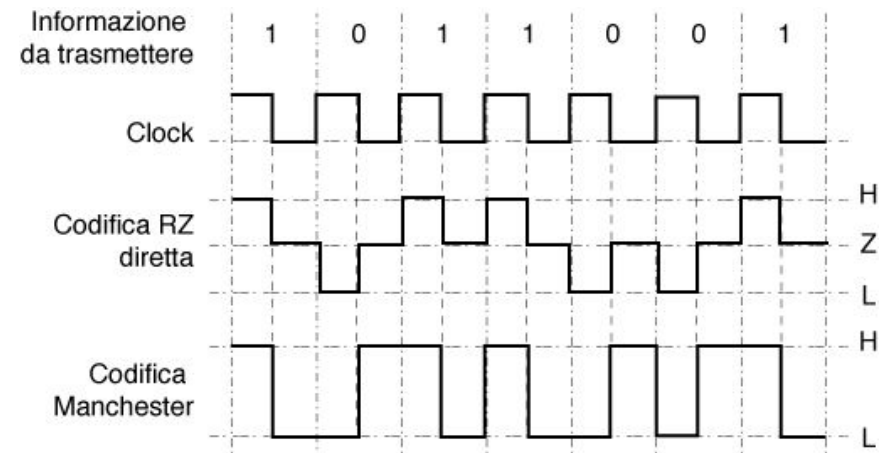
sono le codifiche più semplici; entrambe **“non polari”**:  
l'assenza di trasmissione è assegnata a uno dei due livelli



codifica **RZ**: seconda metà del bit (Clock 2) sempre al livello Z;  
ri-sincronizzazione facilitata, ma si usa doppia banda di frequenza



**“polare”**: il livello Z è associato alla non trasmissione; le codifiche **NRZ** (No Return to Zero) necessitano di una sola tensione di alimentazione (circuito TTL con +5V) ma hanno problemi in caso di perdita di sincronismo



codifica **Manchester** (2 livelli): seconda metà indica transizione (a Z);  
per lunghe sequenze di bit identici, problemi di sincronizzazione  
⇒ codice Manchester differenziale

# Livello fisico della rete

Standard di connessione elettrica/meccanica tra host e linea di trasmissione

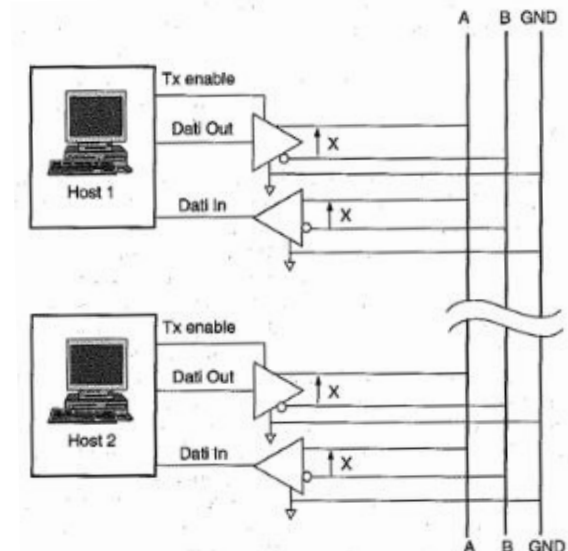
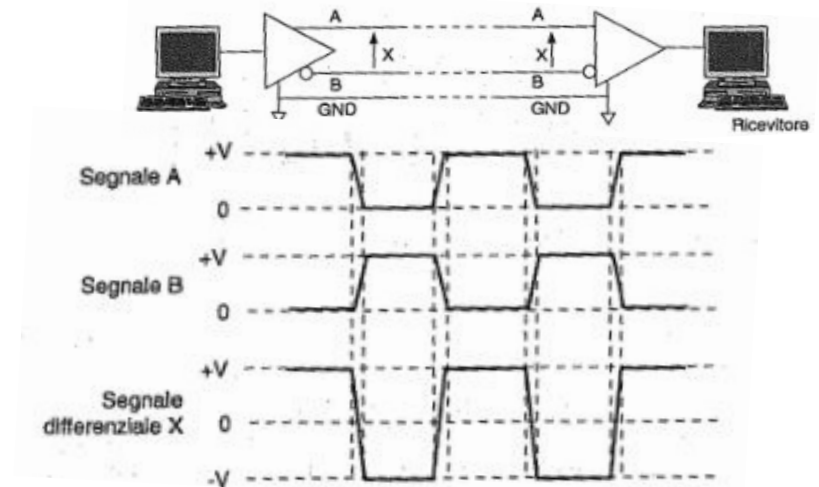
## ❑ IEEE RS232 (seriale, tra i più consolidati)

## ❑ IEEE RS422

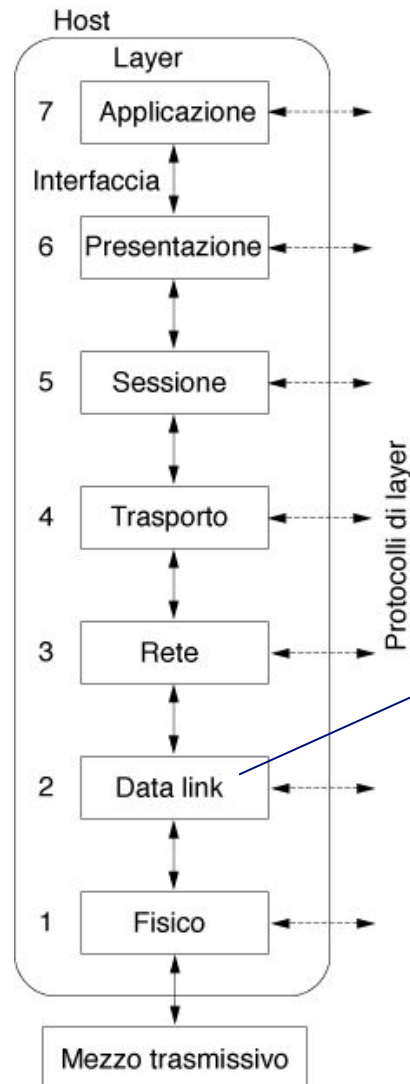
- ➔ ampiamente in uso in ambito industriale
- ➔ segnale negato  $B = -A$  e poi differenziale  $X = A - B$ 
  - range del segnale raddoppiato: più immune ai disturbi
- ➔ 10Mbps, fino a 1200 metri di distanza
- ➔ intrinsecamente simplex
  - (due coppie di cavi + massa per avere full duplex)
- ➔ **1 solo** trasmettitore e fino a 10 ricevitori
- ➔ oppure, più comune è l'uso punto-punto

## ❑ IEEE RS485

- ➔ ampiamente in uso in ambito industriale
- ➔ stessa logica differenziale del RS422
- ➔ **più** trasmettitori e ricevitori (fino a 32)
- ➔ ma solo **1 attivo** per volta (con TX enable)
  - gli altri hanno il circuito di ingresso al canale in uno stato "ad alta impedenza" = come se fossero scollegati
  - occorre gestire i conflitti di accesso



# Architettura ISO-OSI



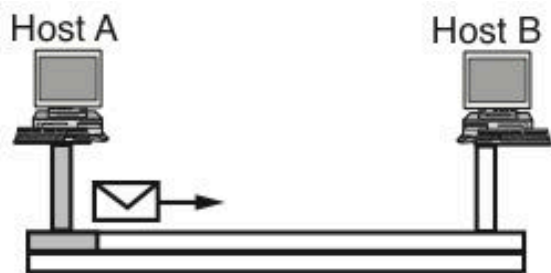
- ❑ stabilisce un collegamento “affidabile” tra due nodi connessi direttamente a livello fisico
  - rileva e, se possibile, corregge errori avvenuti nella comunicazione tra layer fisici
- ❑ esempio: PPP (Point-to-Point Protocol) di TCP/IP

**data link layer:** nel trasmettitore divide il dato in frames, nel ricevitore spedisce gli ACK di ricezione dei frames; effettua il controllo del flusso (velocità di trasmissione); in reti broadcast **gestisce l'accesso al mezzo (MAC)**

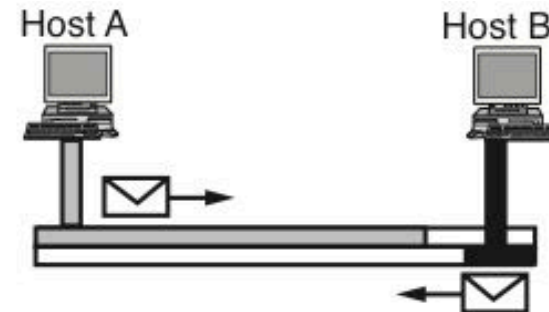
# Gestione dinamica accessi (MAC)

Sistema a collisione: Carrier Sense Multiple Access Collision Detection (**CSMA-CD**)

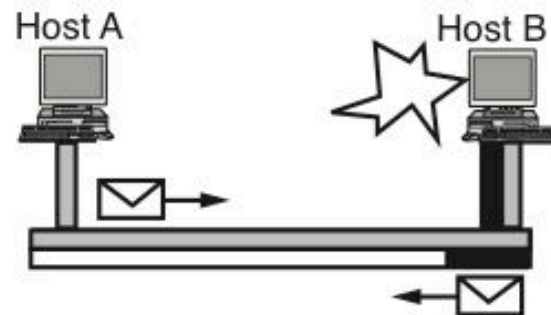
- ogni nodo è sempre in ascolto del canale
- il nodo che trasmette confronta il dato sul canale con quello trasmesso (corrotto o meno)
- rilevata la collisione il nodo attende un tempo casuale e poi ritrasmette



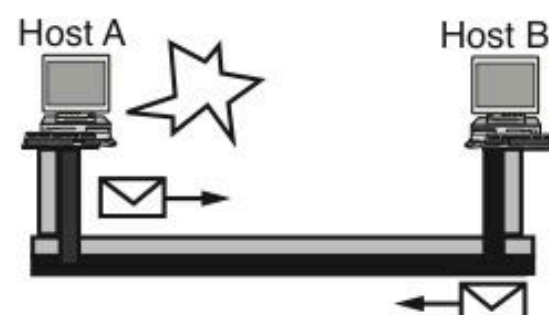
1)  $t=0$ : A inizia a trasmettere dato che il canale è libero



2)  $t=\tau-\varepsilon$ : B inizia a trasmettere e i due messaggi collidono



3)  $t=\tau$ : B rileva la collisione ricevendo la versione corrotta del suo messaggio

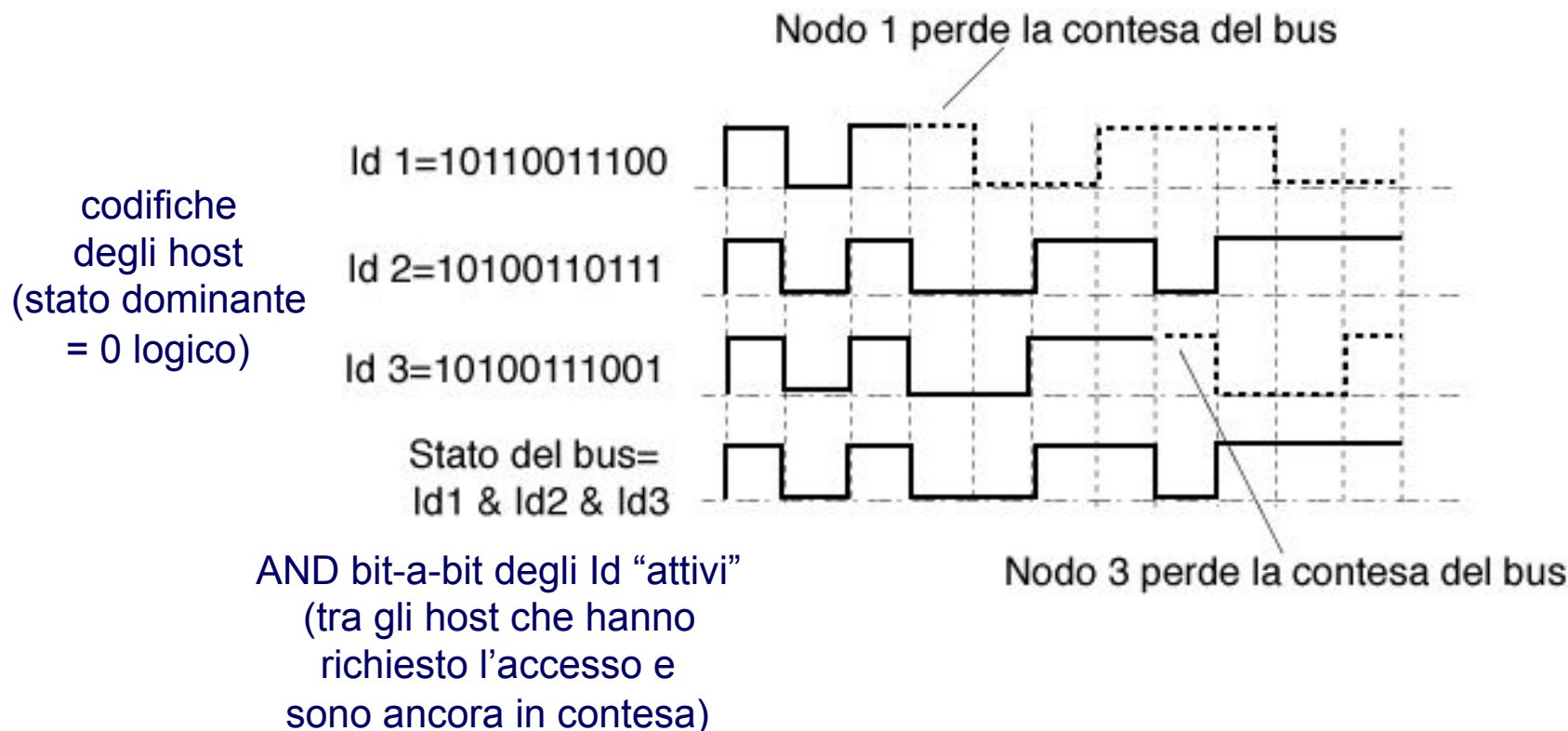


4)  $t=2\tau$ : A rileva la collisione ricevendo la versione corrotta del suo messaggio

# Gestione dinamica access (MAC)

Sistema a collisione: Carrier Sense Multiple Access Collision Resolution (**CSMA-CR**)

- come sopra
- ... ma esiste uno stato fisico dominante del canale
- l'host (nodo) che ha trasmesso l'informazione corrispondente allo stato dominante vince la collisione e continua a trasmettere (ad es.: stato zero in un AND)
- occorre garantire che un solo host sopravviva...

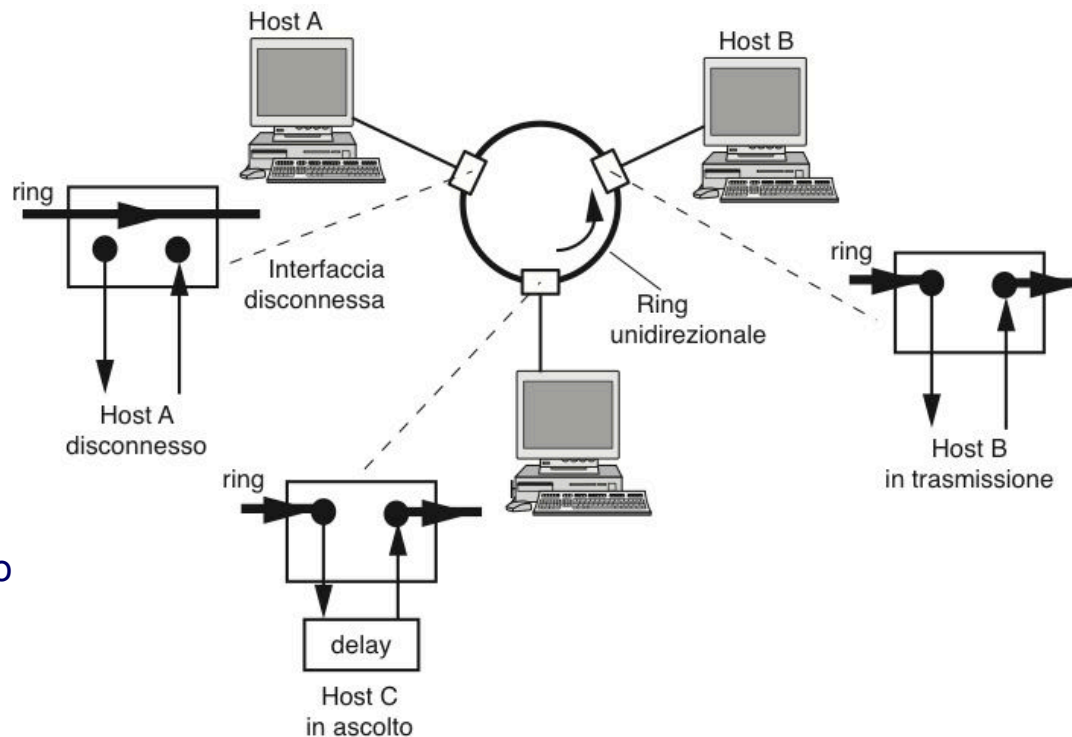




# Gestione dinamica accessi (MAC)

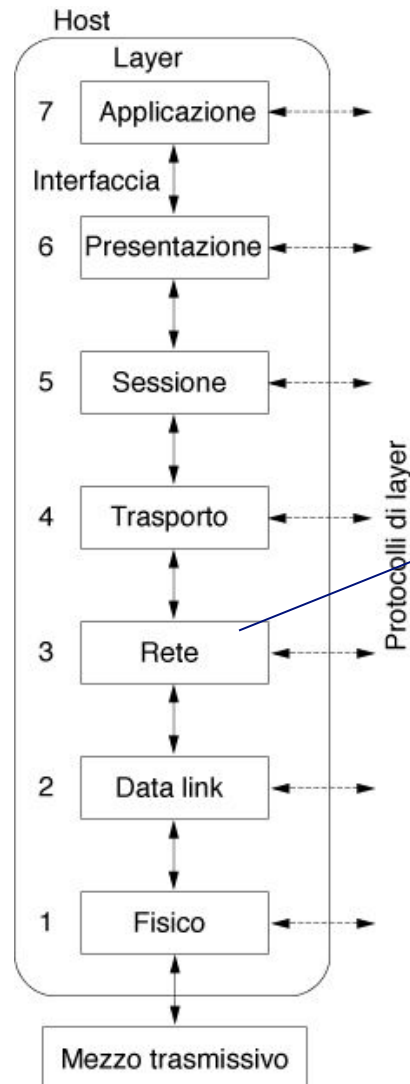
## Sistema ad assenza di collisioni: **Token Bus/Ring** (IEEE 802.4/.5)

- il token circola continuamente ( $\Rightarrow$  stesso tempo deterministico tra due successive interrogazioni del canale da parte di ogni host)
- un nodo trasmette quando è in possesso dell'abilitazione data dal token "originale": lo modifica e rispedisce in circolo assieme al frame del dato (e con un host destinatario)
- la presenza del token "modificato" non permette la trasmissione agli altri nodi
- il ritorno del token modificato al nodo trasmettente indica l'avvenuto successo della trasmissione: il token originale viene rimesso in circolo e altri nodi potranno quindi trasmettere



IBM Token ring ha avuto  
un grosso successo  
commerciale

# Architettura ISO-OSI

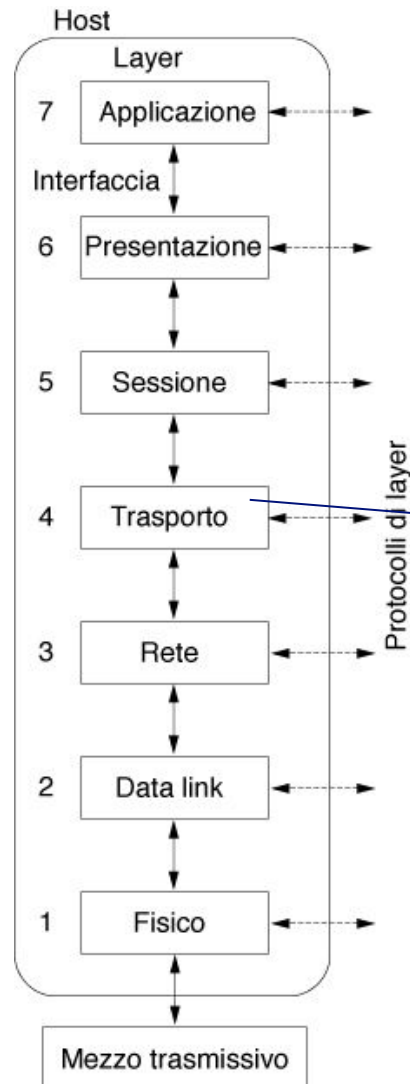


- stabilisce funzioni e procedure per trasferire sequenze di dati di *lunghezza variabile* (pacchetti) tra nodi della *stessa rete*
  - ogni nodo della rete ha un indirizzo unico
- include algoritmi e tabelle di routing, di gestione di gruppi multicast, di assegnazione di indirizzi ai nodi della rete, ..

**network layer:** indirizzamento nelle subnet;  
decisione del percorso (**routing**) da host a host



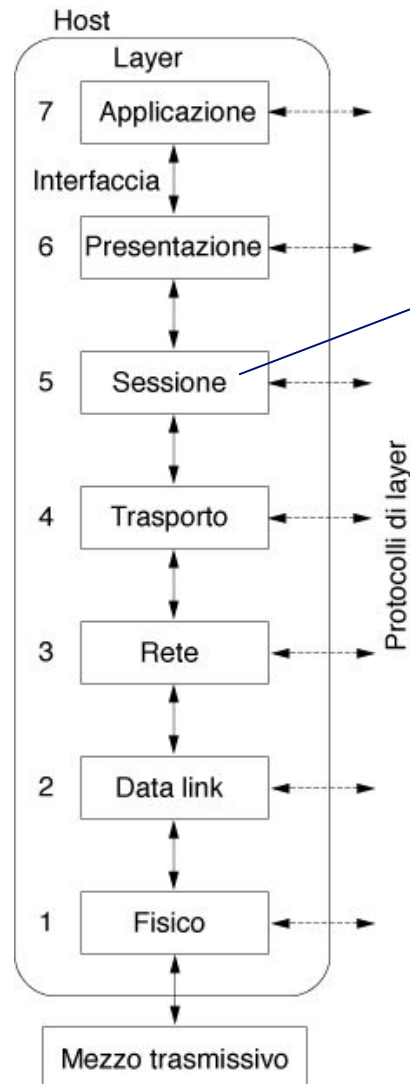
# Architettura ISO-OSI



- invio “affidabile” di segmenti/pacchetti (messaggi) di dati tra nodi (iniziale e finale/i, con indirizzi) della rete
- il controllo di consistenza può essere realizzato con diversi livelli di completezza
  - fino al livello error-free (reinvio dopo timeout)
- esempio: TCP nel protocollo Internet standard

**transport layer:** divisione/ricostruzione di segmenti di dati in/da pacchetti; controllo di consistenza (vs. errori)

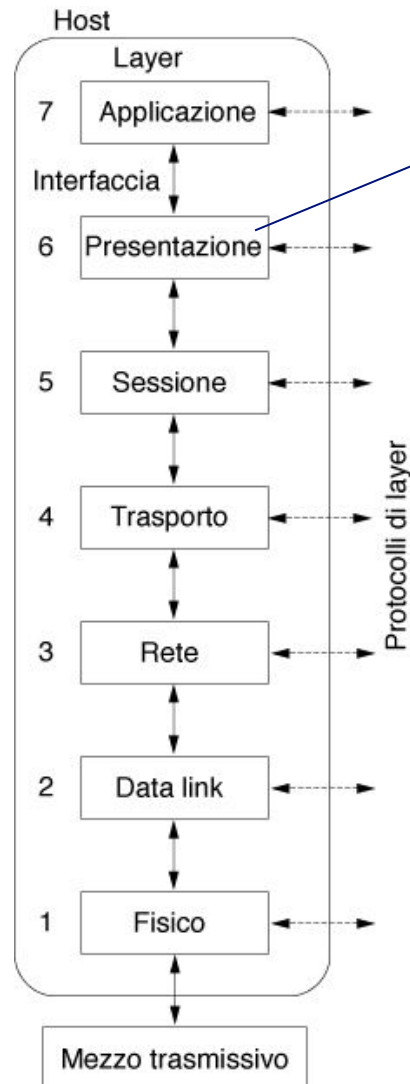
# Architettura ISO-OSI



**session layer:** gestione delle connessioni tra host;  
controllo dei conflitti

- regola i dialoghi tra i dispositivi e/o computer
- inizia, gestisce e termina le connessioni (sessioni) tra applicazioni locali e remote
  - opera in modo full-duplex, half-duplex, o simplex
  - inserisce checkpoint (riavvio dopo malfunzionamenti), gestisce token, ...

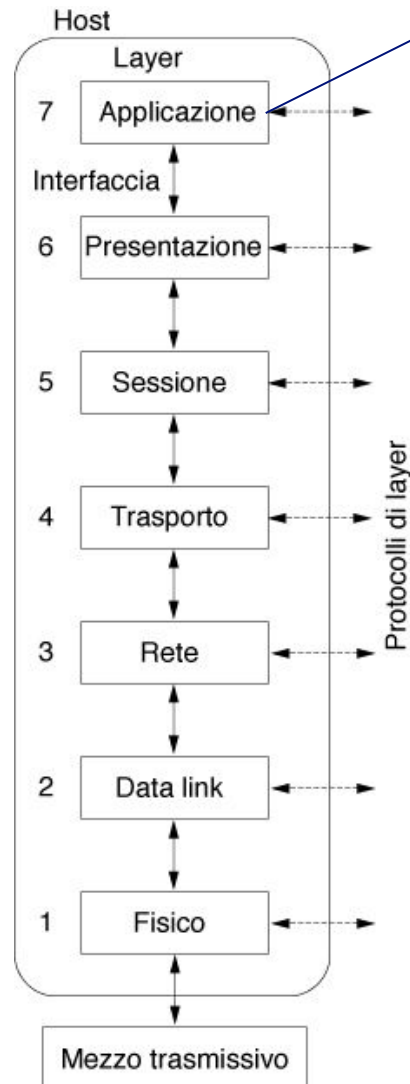
# Architettura ISO-OSI



**presentation layer:** codifica e conversione dei dati, inserimento dei bit di controllo, verifica della correttezza

- fornisce il “contesto” tra entità dell’application layer
  - mappatura tra sintassi e semantiche differenti usate dallo strato superiore
  - a volte chiamato anche “syntax layer”
- permette l’indipendenza dalla rappresentazione dei dati, traducendo tra diversi formati applicativi e di rete
  - in/da codice ASCII, in/da XML, ...
  - crittografia, compressione, ...

# Architettura ISO-OSI



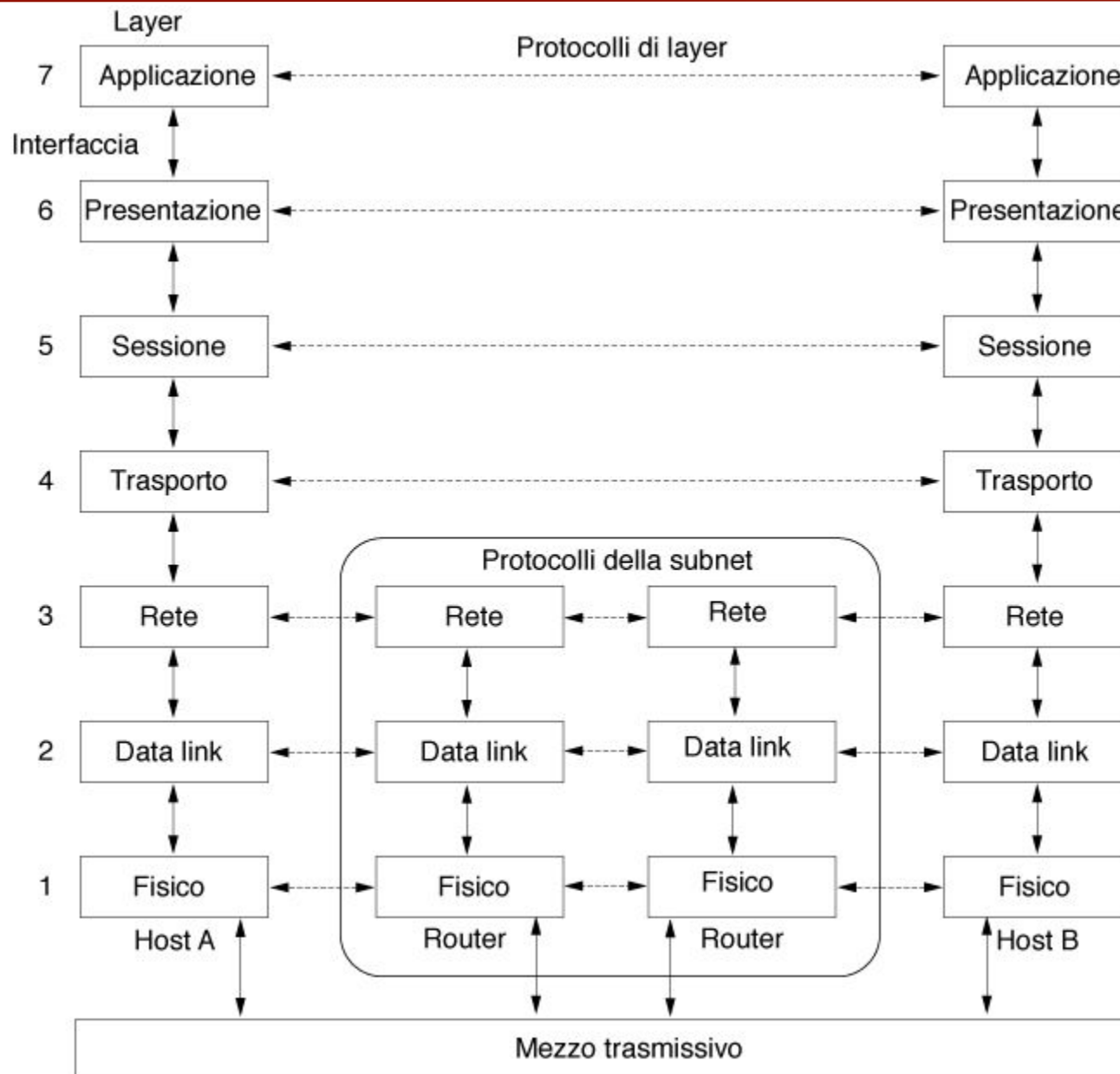
**application layer:** protocolli necessari all'utilizzatore finale (HTTP, POP, FTP, SMTP ...)

- verifica l'identità tra gli host che comunicano, determina la disponibilità delle risorse necessarie e sincronizza la comunicazione

N.B: protocolli e layer del modello TCP/IP di Internet sono deliberatamente meno rigidi del modello OSI

- application layer (resto del 5, 6-7)
- transport layer (4, parte del 5)
- internet layer (resto del 3)
- link layer (1-2, parte 3)

# Architettura ISO-OSI



**utilizzo finale del dato**

**codifica e preparazione dato**

**gestione connessioni**

**divisione/ricostruzione  
dati complessi**

**funzioni di routing**

**accesso al mezzo**

**gestione del supporto fisico**

**connessione di due host  
e due router in rete WAN**

# Comunicazione real-time: specifiche

---

## Comunicazione tra task real-time

### ❑ messaggi periodici

- ➔ generati o utilizzati da task periodici
- ➔ es: letture sensori, comandi di controllo agli attuatori

### ❑ messaggi aperiodici

- ➔ generati o utilizzati da task aperiodici
- ➔ es: comandi operatore

### ❑ messaggi sporadici

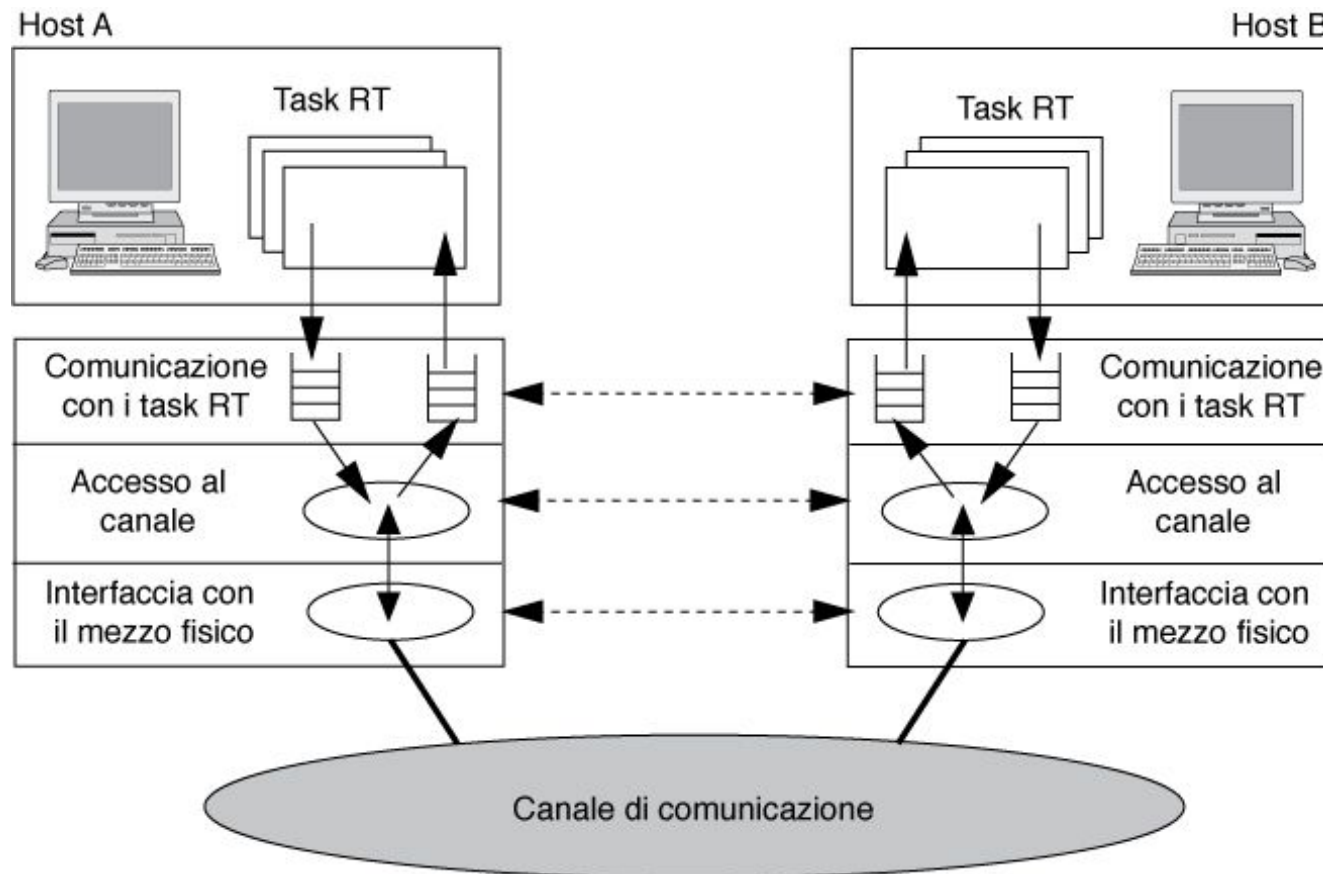
- ➔ generati da task aperiodici con vincoli hard real-time
- ➔ es: allarmi

# Comunicazione real-time: specifiche

## Comunicazione tra task real-time

- ❑ missed packet: pacchetti non arrivati in tempo
- ❑ lost packet: pacchetti persi (ad es., per evitare overflow dei buffer si scartano pacchetti verso task non ready)
- ❑ delay: ritardo di comunicazione, tipicamente aleatorio  $\Rightarrow$  jitter

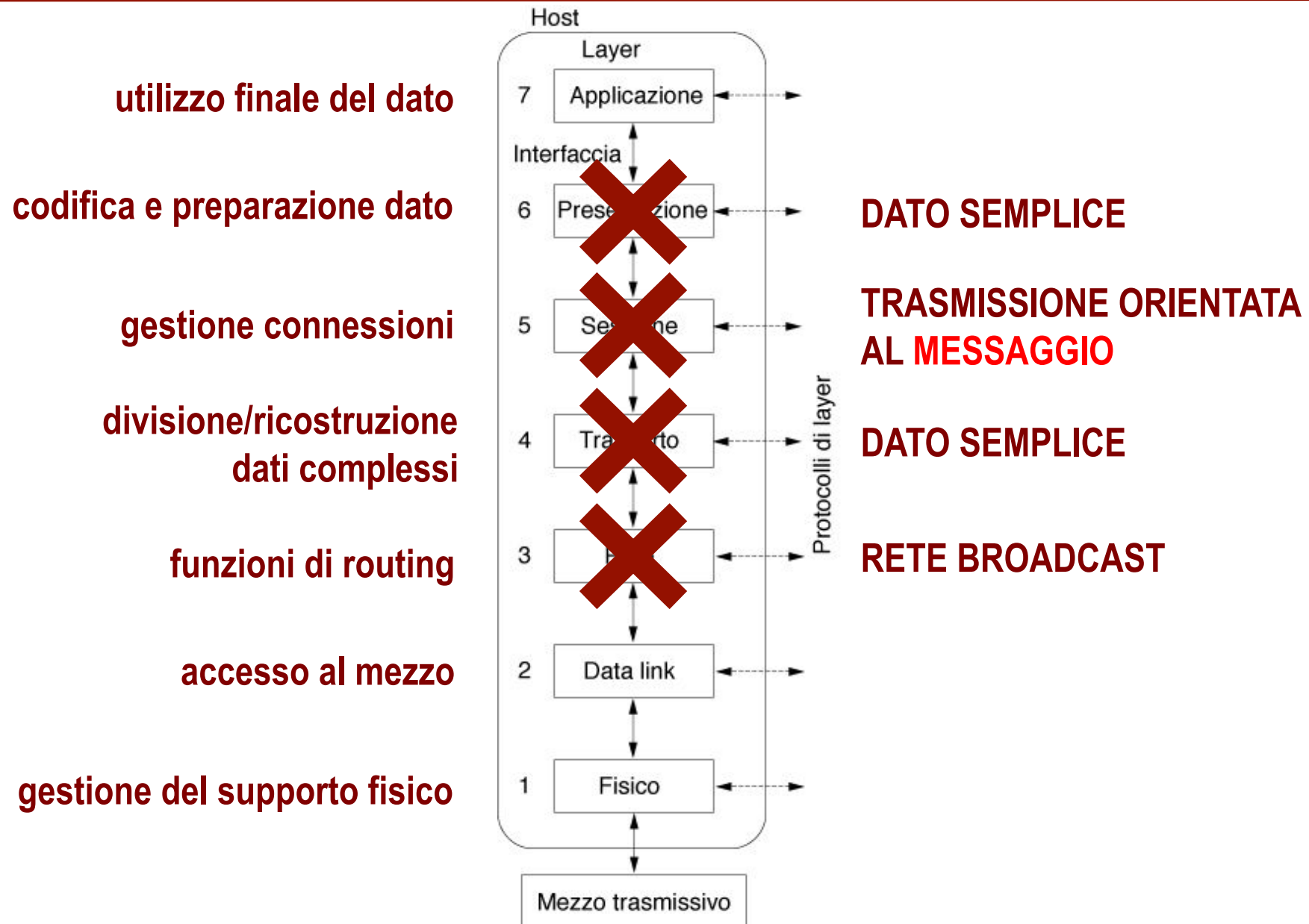
qui: uso di  
**algoritmi di  
scheduling**  
in real time

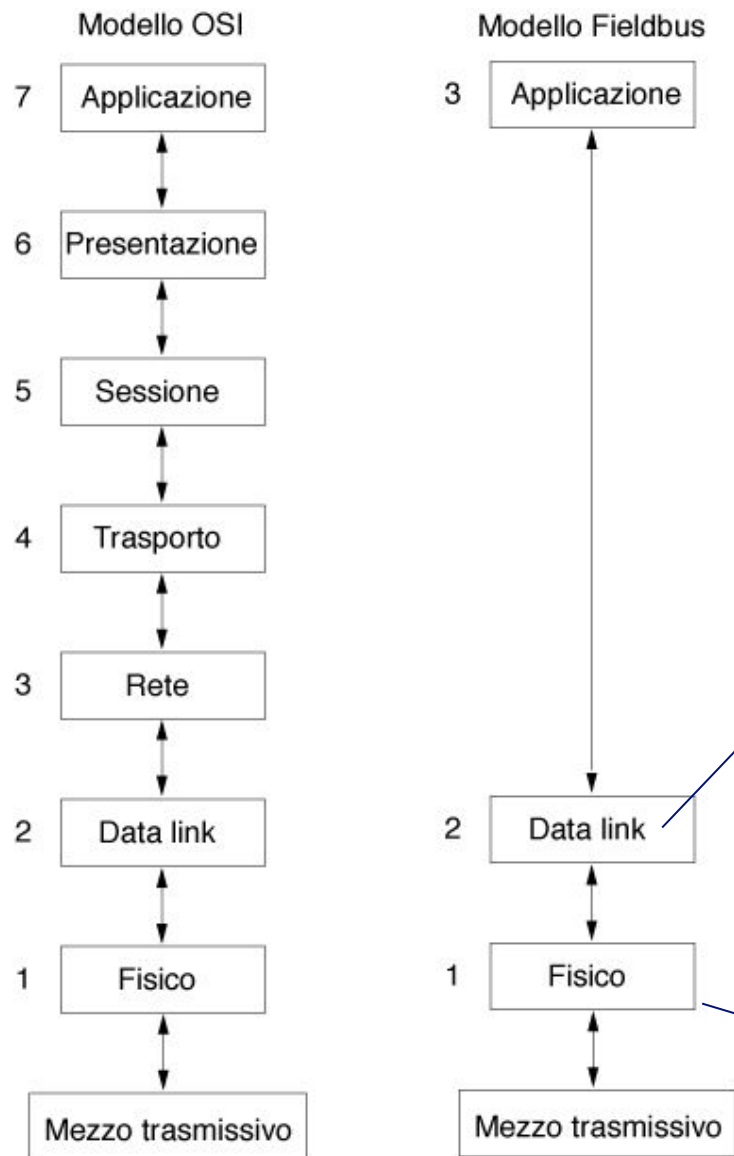


- ❑ la struttura con 7 livelli del modello OSI è tipicamente troppo onerosa per applicazioni con bus di campo
  - ➔ overhead eccessivo, anche considerando che le dimensioni dei singoli messaggi è tipicamente piccola
  - ➔ molti servizi dei livelli intermedi non sono necessari
  - ➔ grande attenzione ai metodi di accesso al mezzo trasmissivo (MAC) che impattano sulle caratteristiche temporali dei messaggi scambiati



# Riduzione dell'architettura ISO-OSI



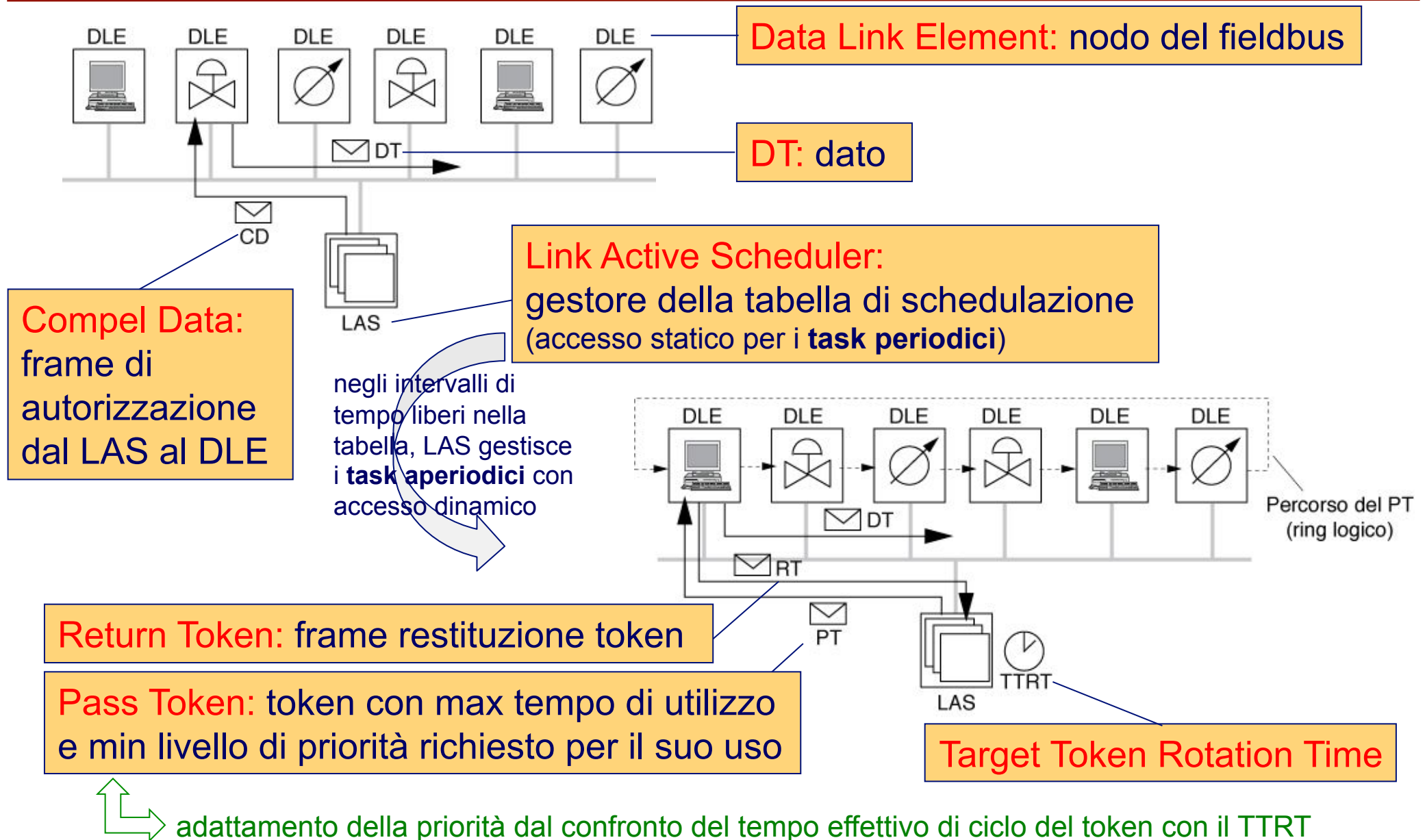


## data link layer:

- il punto critico è la gestione dell'accesso al mezzo (MAC) con vincoli real-time
- tipicamente si usa un protocollo
  - a pre-schedulazione per i task periodici
  - a token per i task aperiodici
- il coordinamento dei due metodi è effettuato dal Link Active Scheduler

## physical layer:

- funzioni di interfaccia dipendenti (Data Terminal Equipment) o indipendenti (Data Communication Equipment) dal mezzo fisico



# Bus di Campo: norme IEC

- ❑ IEC 62026 (07/2000): controller/device interface
  - ➔ Smart distributed systems, AS-I, DeviceNet
- ❑ IEC 61158 (04/2003): fieldbus standard for use in industrial control systems (real time, a livello di campo)
  - ➔ regolamento del livello fisico
  - ➔ regolamento del livello dati
  - ➔ regolamento del livello applicazione
- ❑ IEC 61784 (05/2003): definizione delle communication profile family
  - ➔ Fieldbus Foundation (1994)
  - ➔ Controlnet (1997)
  - ➔ Profibus (1996)
  - ➔ P-net (1990)
  - ➔ WorldFIP (1993)
  - ➔ Interbus (1998)
  - ➔ Swiftnet (2000)
- ❑ CanOpen, ModBus ecc: bus di campo diffusi ma non ancora normati

