RETI LOCALI (LAN)

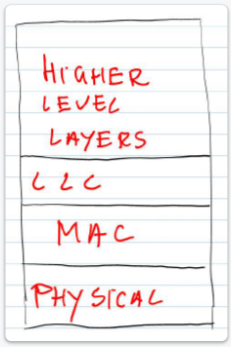
Servizi di connessione contingentati verso un'area locale, cioè geograficamente non estesa

- Pensato decine di anni fa
- Vedi reti ring, bus, star, etc... (introduzione corso)
- Specifica livelli bassi (1 e 2)
- Si suppone che i dispositivi interni alla rete abbiano a disposizione *mezzi trasmissivi di qualità*
  - Il controllo dell'integrità (che nella struttura OSI si effettua a livello collegamento su base link-to-link), qui viene inserita agli estremi (edge - base end-to-end) ed è gestita dall'LLC (vedi dopo)
- Standard riferimento delle reti locali: IEEE 802, in particolare diremo

$$\text{IEE } 802.xxx \quad , \quad xxx \rightarrow \text{servizio da specificare}$$

Architettura Protocollare 802

Fatta a strati (livelli)



Higher Layer → non sono specificati (indica che lì ci saranno i livelli più alti in TCP)
LLC → Logical Link Control
MAC → Medium Accesso Control
Fisico → livello più basso

LLC (LOGICAL LINK CONTROL)

Strato comune a tutti gli standard IEEE introdotti (a differenza del MAC che è caratteristico delle particolari reti LAN che esamineremo)
Ha il compito di verificare su base *end-to-end* *l'integrità dei dati* ricevuti

- Non crea problemi come concetto se stiamo utilizzando mezzi "di fiducia", di qualità (come supposto all'inizio)

Definisce uno standard *unico* per tutte le reti LAN della famiglia IEEE 802

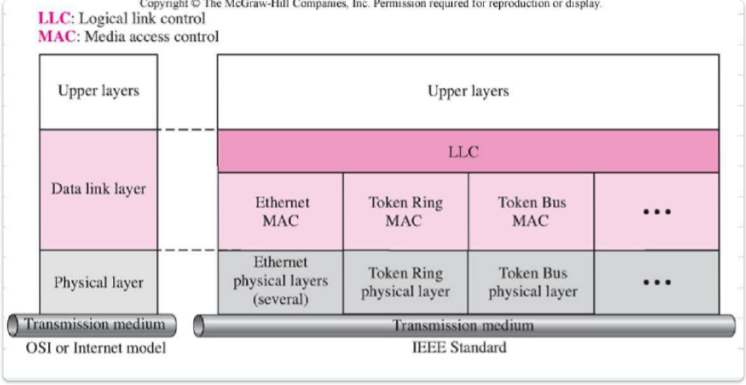
- cioè va bene per tutte le tipologie di reti (LAN)

Gestisce/implementa le seguenti modalità:

- Connection less non affidabile (senza riscontro)
  - Modalità detta logical data link
- Connection less affidabile (con riscontro)
  - Modalità detta logical data link alternativo
  - In caso di errori si richiede il re-invio
- Connection Oriented
  - Modalità detta data link connection

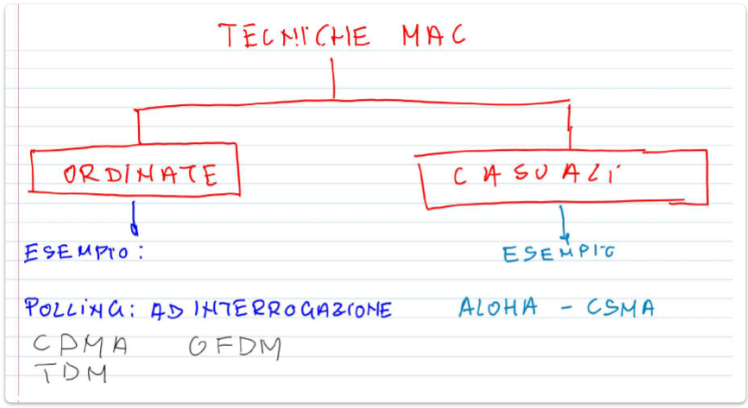
CONFRONTO OSI vs IEEE

- Si noti come LLC sia lo stesso
- MAC invece specifica a seconda della topologia della rete (ethernet, ring, bus...)



MAC

Livello con la caratteristica di avere la tecnica di *condividere l'accesso a uno stesso mezzo fisico* a più utenti (in istanti diversi) utilizzando una tecnica specifica.
Esistono due grandi famiglie di queste tecniche:



- **Casuali**: il primo utente che accede al dispositivo ne diventa il possessore, senza preoccuparsi degli altri utenti che potenzialmente potrebbero accederci. Possibili collisioni (pensiamo se due utenti fanno richiesta contemporaneamente a una risorsa, o se un utente chiede l'accesso a una risorsa già occupata)
  - Tecniche di accesso casuale (cfr. approfondimenti dopo):
    - ALOHA
    - CSMA (usato in reti Ethernet)
- **Ordinate**: si stabilisce una regola che vale per gli utenti autorizzati ad accedere al servizio/dispositivo (esempio: schedulare una tabella per gestire i tempi di accesso circa l'utilizzi di un PC). In questo modo si evitano conflitti (collisioni)
  - Tecniche di accesso ordinato:
    - **FDMA***, **TDMA** (accesso multiplo a divisione di frequenza o tempo: simile ad esempio all'FDM/TDM, che divideva la banda/tempo_utilizzo di un canale in canali e ciascuno viene assegnato a un utilizzatore)
    - *CDMA* (Code Division Multple Access) → utilizzata oggigiorno soprattutto (più complessa): sfruttano i segnali ortogonali e la loro correlazione $\int_0^t x(t)y(t)\,dt = 0$. Basterà allora assegnare a ogni utente un segnale che rispetta tale proprietà con tutti gli altri. In ricezione soltanto quando si considera l'integrale con la copia esatta del segnale che vogliamo estrarre avremo valore diverso da zero (gli altri saranno nulli). In questo modo abbiamo estratto ciò che volevamo (nota: è un caso ideale)
    - **POLLING**: scheduling ordinato secondo un criterio ben definito (vedi dopo)

TECNICHE ORDINATE

POLLING (interrogazione)

Prevede invio di un messaggio con struttura nota che *abilita chi lo riceve* ad accedere al canale

- Per evitare conflitti sull'accesso è necessario prevedere una mobilità di invio del messaggio adatta, che abiliti cioè un solo (specifico) utente del gruppo

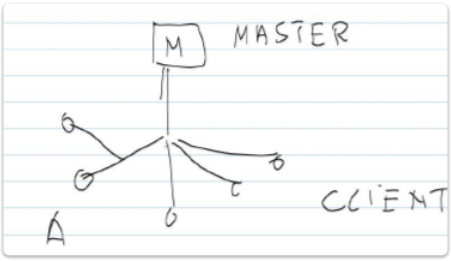
Ci sono varie modalità per effettuare ciò, ad esempio:

ROLL-CALL

Tecnica di tipo *centralizzato* - esiste una **entità** che si occupa di coordinare l'accesso di ne fa richiesta in modo ordinato

- Come quando l'insegnante recita l'appello: ogni utente viene interrogato seguendo un ordine prefissato (alfabetico ad esempio)
- Così fa l'entità centrale: invia il messaggio di autorizzazione al primo user che ha nella lista (del database). L'utente riconosce il messaggio, lo trattiene e accede al canale. Terminata la fase di accesso il terminale dell'utente rilascia l'autorizzazione e lo comunica alla entità centrale

rilasciando il messaggio che aveva trattenuto. Si procede quindi con le successive interrogazioni (se un utente non ha informazioni da riferire, rilascia subito il messaggio)

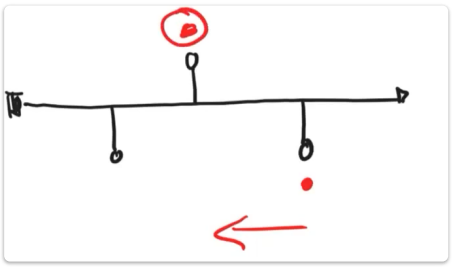


😡: tempo di non utilizzo della rete (tipo quando un utente poi non richiede il servizio quindi rilascia subito il messaggio), delay dei tempi di rilascio e tempo interrogazioni varie
> Tecnica alternativa: esiste e si chiama Hub-Polling (vedi dopo)

HUB-POLLING

Tecnica di tipo *distributivo* e cooperativo
Idea simile alla struttura a bus:

- L'entità centrale (Master) chiama la stazione più lontana
- La stazione riceve l'autorizzazione e accede, rilasciando il messaggio
- Il messaggio viene rilasciato dalla stazione immediatamente più vicina alla precedente
- Continua il "passaparola" fino a tornare al Master



In questo modo si *riducono al massimo i tempi di latenza*

MODALITA' DI ACCESSO (vedi lezione 5.4.22 minuto 50)

Due modalità:

- **Gated** (limitato): non si riferisce ad un tempo di accesso costante e fisso. Il tempo di accesso è infatti *relativo* alla trasmissione dei pacchetti arrivati al nodo nell'intervallo di tempo compreso tra l'istante di arrivo dell'autorizzazione ad un ciclo e l'istante di arrivo al ciclo successivo
  - L'intervallo di riferimento entro il quale si considerano tutti i pacchetti arrivati non è prefissato, ma è dato da viene mandato il messaggio di autorizzazione al primo client a quando finalmente ritorna al giro successivo
  - Si consente quindi l'accesso per un tempo limitato
- **Esaustivo**: prevede che un utente rilasci l'accesso solo quando non ha più pacchetti da trasmettere. Si inviano i pacchetti arrivati da quando si rilascia l'autorizzazione a quando arriva nuovamente oltre ai pacchetti dati che arrivano durante la fase di accesso
  - Si trasmette tutto e ci si ferma solo quando il buffer è vuoto
  - Non si introduce quindi alcuna limitazione