



Lezione 31

Reti di calcolatori

Proff. A. Borghese, F. Pedersini

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano

Introduzione

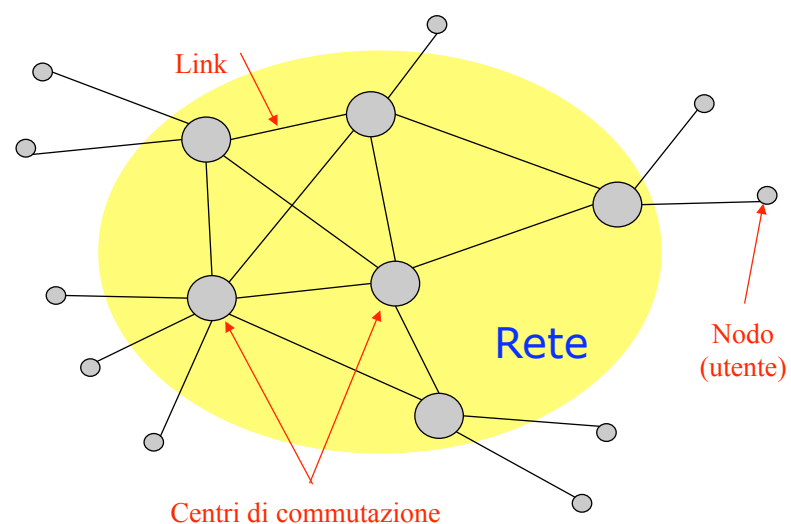


❖ Definizione (Tanenbaum: "Computer Network")

- Rete di calcolatori = insieme di nodi di calcolo indipendenti e interconnessi
 - ♦ Es. rete BANCOMAT non è una "computer network"

❖ Esempi

- LAN
- Rete wireless
- Internet





- ❖ Non esiste una tassonomia universalmente accettata delle reti.

Classificazione per...

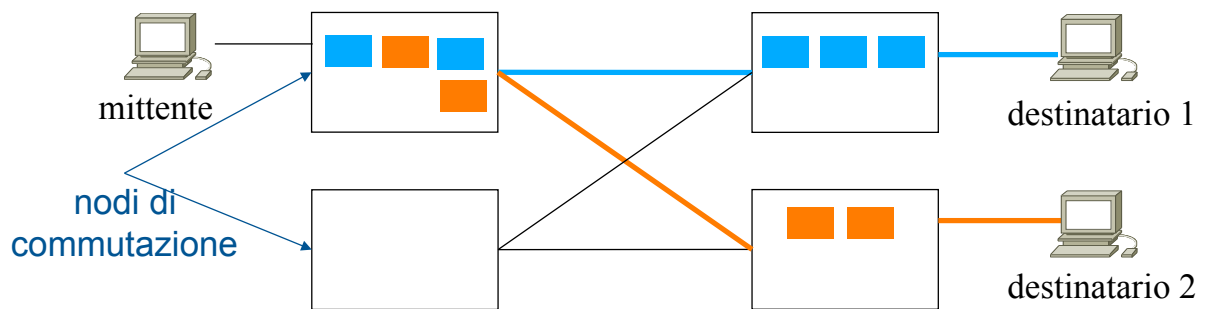
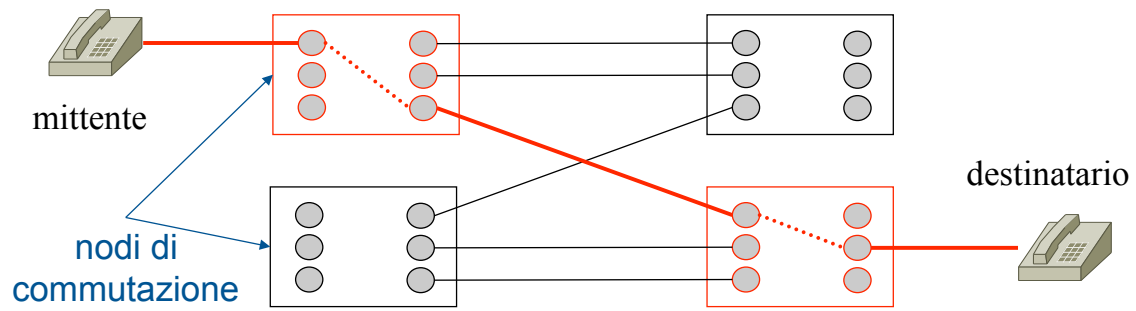
- ❖ **...dimensione**
 - bus, LAN, MAN, WAN, Internet
- ❖ **...topologia**
 - bus, ring, a stella,
- ❖ **...modalità di collegamento**
 - broadcast, peer-to-peer, ...
- ❖ **...tipo di traffico**
 - commutazione di circuito, commutazione di pacchetto



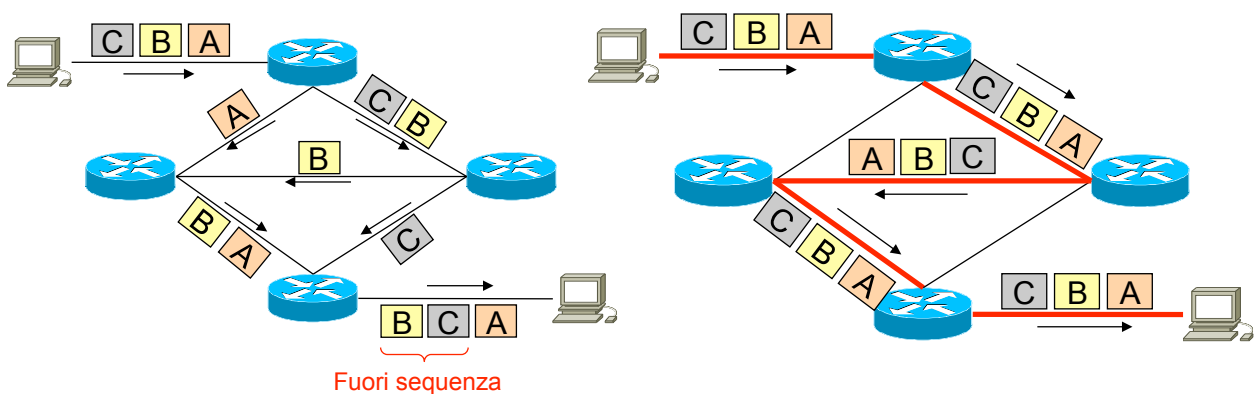
| Distanza tra nodi | "Taglia" della rete |
|-------------------|---------------------|
| 0.1 m | Circuit board |
| 1 m | System |
| 10 m | Room |
| 100 m | Building |
| 1 km | Campus |
| 10 km | City |
| 100 km | Country |
| 1000 km | Continent |
| 10000 km | Planet |

- Micro-computer
- Multi-computer
- } Local Area Network (LAN)
- } Metropolitan Area Network (MAN)
- } Wide Area Network (WAN)
- Internet

Commutazione di circuito/pacchetto



Commutazione di pacchetto



- ❖ **Datagramma:** ciascun pacchetto è entità **autonoma**, instradata in modo indipendente in base all'indirizzo contenuto nell'intestazione.
- ❖ **Circuito virtuale:** mediante uno scambio di pacchetti di controllo si stabilisce una **connessione virtuale** (*non dedicata*) lungo la quale verranno instradati tutti i pacchetti dati.



- ❖ **Scambio files (FTP) tra PC, via Ethernet:**
 - Devo definire:
 - Il collegamento fisico tra i nodi terminali
 - ✦ Come viaggia l'informazione? → Mezzo trasmissivo
 - Il meccanismo di comunicazione dati
 - ✦ Tecnica di trasmissione digitale sfruttando il MT
 - Le regole di dialogo tra i vari terminali
 - ✦ Quando parlo? Quando ascolto? → Protocollo di accesso (MAC)
 - Meccanismo che si prende in carico i dati da spedire/ricevere e li invia sulla/raccoglie dalla rete.
 - ✦ Driver della scheda di rete / Windows Socket
 - Interfaccia utente "trasparente" di scambio files
 - ✦ Applicazione (software) sul terminale utente
- ❖ Perché la rete funzioni devo progettare/gestire/sincronizzare tutti questi aspetti → sistema complesso



Per gestire sistemi complessi è necessario:

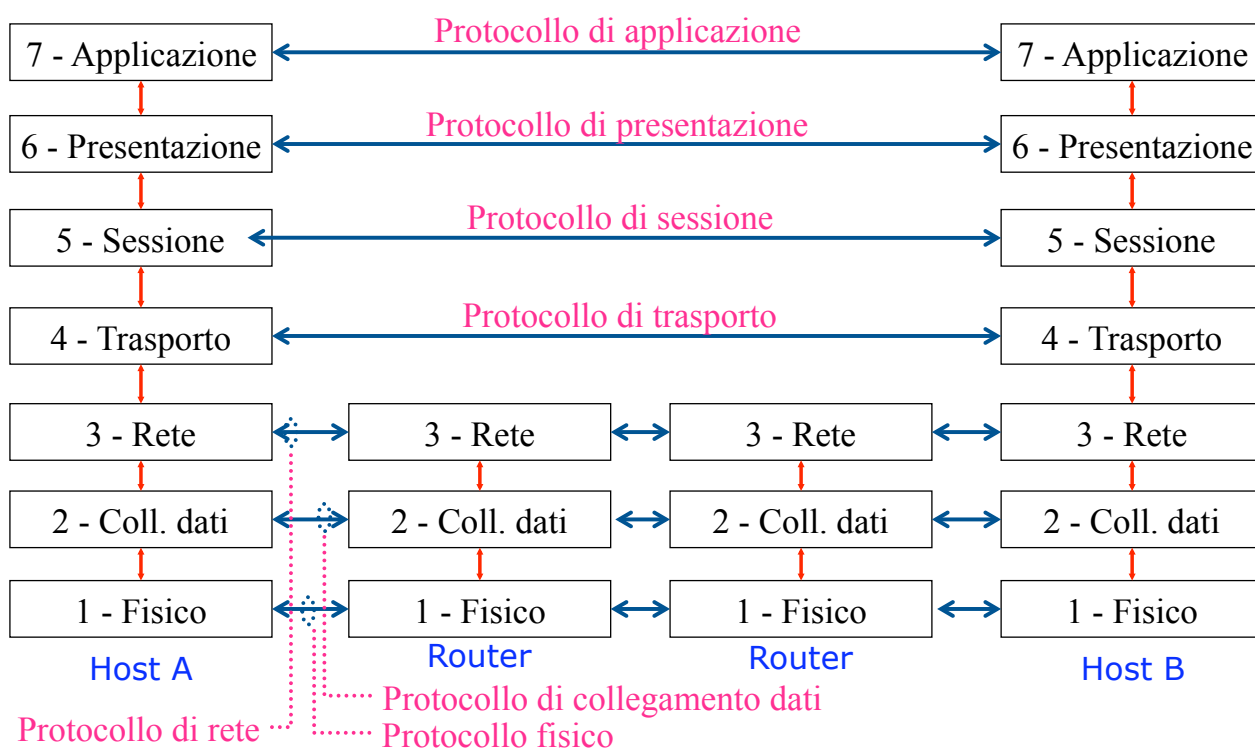
1. STANDARDIZZARE
2. Dividere il problema in sottoproblemi

→ **Modelli di rete definiti a LIVELLI**

- ❖ Definisco singolarmente, per ogni livello:
 - il suo funzionamento → Protocollo
 - il suo modo di interagire con i livelli adiacenti (sopra e sotto) → Interfaccia
- ❖ **Modelli standard di rete, a livelli:**
 - **TCP-IP** (4 livelli)
 - **ISO-OSI** (7 livelli)



Lo stack ISO-OSI



Modello ISO-OSI: i 7 livelli

ISO-OSI:

1 - Fisico

- Definizione del mezzo trasmissivo (MT). Fornisce al livello 2 un collegamento tra nodi sul quale vengono trasmesse le unità dati

2 - Collegamento dati

- Definisce la tecnica di trasmissione dei dati sul MT

3 - Rete

- ha come obiettivo primario l'instradamento (strada da mittente a destinatario)

4 - Trasporto

- fornisce al livello superiore un servizio affidabile di *trasferimento dell'informazione*

5 - Sessione

- gestisce ogni singolo "dialogo" instaurando il collegamento, negoziando le regole di comunicazione e chiudendo il collegamento.

6 - Presentazione

- si preoccupa della coerenza sintattica dell'informazione fornita al livello 7

7 - Applicazione

- gestisce la comunicazione a livello di applicazione sw (es. Telnet, FTP, Web)



TCP-IP:

1-2: Fisico/Collegamento dati: "Host-to-network"

- mezzo trasmissivo e metodo di accesso al mezzo e trasmissione dati associato

3: Rete: IP

- Indipendente dal mezzo trasmissivo → Anima di Internet

4: Trasporto: TCP (/UDP)

- Sfruttano IP per fornire servizi diversi (connection oriented / connectionless) alle applicazioni

5-7: Applicazione

- telnet, FTP, e-mail, WWW

Stack TCP/IP su rete locale (LAN)



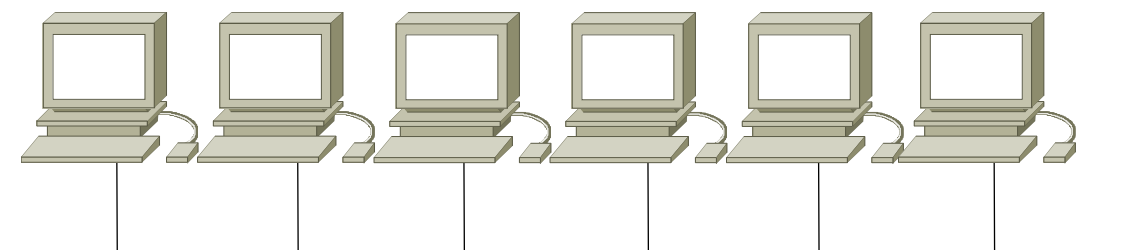
❖ TCP/IP su LAN: caso più diffuso di rete locale

- LAN di Windows, UNIX/LINUX, Mac, ...

Liv. 1: Topologia: bus, mezzo trasmissivo: cavo in rame

Liv. 2: Accesso al MT: **Ethernet (IEEE 802.3)**

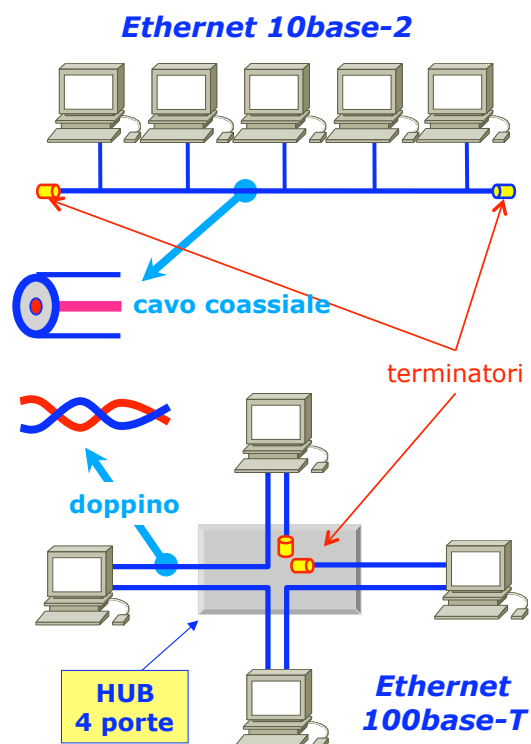
Liv. 3,4: Protocolli **IP, TCP (Internet)**



Liv. 1: il livello fisico in Ethernet (IEEE 802.3)

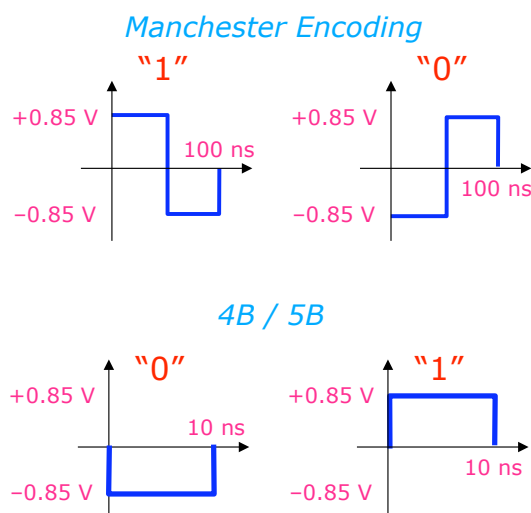
Diversi standard:

- ❖ **Cavo coassiale** (10base-2, 10base-5)
 - Necessario un terminatore per simulare lunghezza infinita
 - Capacità di canale: **10 Mb/sec**
 - $L_{MAX} = 2500$ m
- ❖ **Cavo a doppino** (100base-T)
 - Doppino, pseudo-stella (bus)
 - Capacità di canale: **100 Mb/sec**
 - $L_{MAX} = 100$ m
- ❖ **Fibra ottica** (10, 100base-F)
- ❖ **Gigabit Ethernet** (1000Base-T)
 - Capacità di canale: **1 Gb/sec**
 - $L_{MAX} = 100$ m



Livello fisico in Ethernet: codifica

- ❖ **Tecnica di trasmissione:** come viaggia l'informazione sul mezzo (cavo, fibra, ...)?
- ❖ **10 Mb/sec: Manchester Encoding**
 - autosincronizzante
 - robusto → circuiti semplici
 - ridondanza: **100 %**
- ❖ **100 Mb/sec: 4B/5B** (5 bit per trasmetterne 4)
 - Ogni gruppo di 5 bit contiene almeno 2 transizioni (1/0 o 0/1)
 - meno robusto
 - ridondanza: **25 %**



| Codifica 4B/5B: | | |
|--------------------|-------|-----------|
| Originale (4 bit): | 0000 | 1110 ... |
| Trasmessa (5 bit): | 10101 | 01001 ... |



❖ **Funzioni** svolte dal livello di **collegamento dati**:

- **trasmissione affidabile** sul MT
 - ✦ Sincronizzazione affidabile → **framing**
 - ✦ Controllo errori → **codici controllo errore**
- **Gestione del MT condiviso (MAC)**
 - politica di arbitraggio

→ **Soluzioni**:

- **Framing**:
 - ✦ trasmetto informazione in **pacchetti di lunghezza limitata**
- **Controllo errori**:
 - ✦ allego ad ogni frame informazioni (ridondanza) per rivelare errori di trasmissione
- **MAC: Medium Access Control**
 - ✦ Definisco una **politica di gestione**, che viene rispettata da tutti i nodi collegati

MAC: *protocolli statici e dinamici*



❖ **Protocolli MAC statici**:

- **TDMA: Time-Division Multiple Access**
 - ✦ Il tempo viene ripartito ciclicamente fra i nodi della rete.
- **FDMA/WDMA: Frequency/Wavelength-division Multiple Access**
 - ✦ Il range di frequenza/lunghezze d'onda messo a disposizione dal MT viene suddiviso tra i nodi della rete.

❖ **Protocolli MAC dinamici**:

- **ALOHA**
 - ✦ ogni utente inizia a trasmettere quando vuole
 - ✦ se due o più trasmissioni collidono vengono distrutte
 - ✦ attraverso un meccanismo di feedback gli utenti si accorgono delle collisioni e ritentano la trasmissione dopo un tempo casuale
- **“Carrier Sense Multiple Access” (CSMA)**
 - ✦ Prima di trasmettere ogni utente “ascolta” se la linea è libera.
 - ✦ Periodo di vulnerabilità ridotto al tempo di propagazione.
- **CSMA/CD (CSMA – Collision Detection)**
 - ✦ il trasmettitore resta in ascolto anche durante la propria trasmissione
 - ✦ interruzione della trasmissione se viene rilevata una collisione



MAC su Ethernet: IEEE 802.3

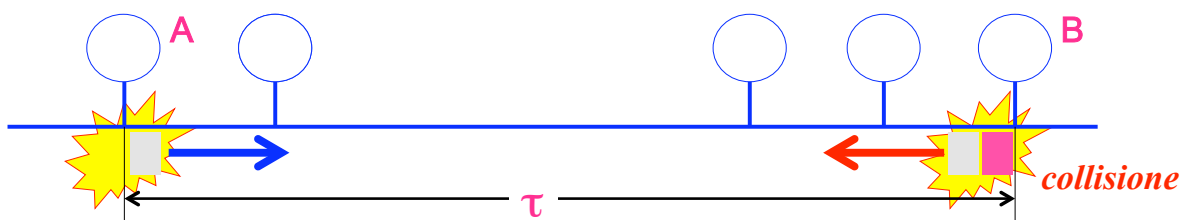
❖ Protocollo CSMA/CD:

1. Prima di trasmettere, ogni utente “ascolta” se la linea è libera
2. Se sì, l’utente inizia la trasmissione, ma resta in ascolto per rivelare eventuali collisioni
3. Se durante la trasmissione viene rilevata una collisione → interruzione della trasmissione → notifica a tutti della collisione → si ritenta dopo un certo tempo.

➤ Per essere sicuro che tutti i nodi notino la collisione, il pacchetto deve durare almeno 2τ

➤ Per $D_{MAX} = 2,5 \text{ km}$, $r = 10 \text{ Mb/s} \rightarrow 2\tau = 51,2 \mu\text{sec} = 512 \text{ bit} = 64 \text{ bytes}$

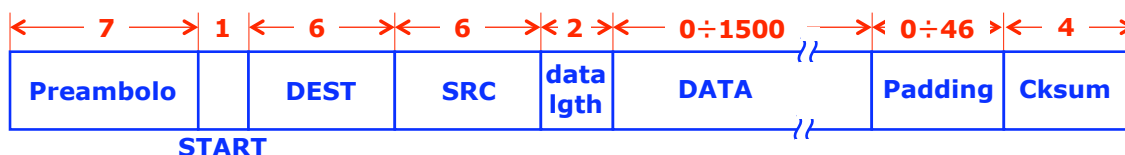
➤ Per far sapere a tutti della collisione, emetto un collision burst (durata $> 2\tau$)



// frame Ethernet

❖ “Pacchetto” di lunghezza variabile: $64 \div 1528 \text{ bytes}$

- **Preambolo:** 7 bytes: **10101010**
- **START:** 1 byte: **10101011**
- **DEST e SRC** – indirizzi mittente e destinatario (**MAC Address:** 6 bytes) unico al mondo per ogni dispositivo Ethernet
- **DATA LENGTH** – lunghezza del campo dati ($0 \div 1500$)
- **DATA** ($0 \div 1500 \text{ bytes}$)
- **PADDING** – ($0 \div 46 \text{ bytes}$) necessario per raggiungere la lunghezza minima di frame di 64 bytes
- **CHECKSUM** – (4 bytes) codice rivelatore di errori



Frame Ethernet



Gestione delle **collisioni** – ritrasmissione

- ❖ Se ho collisione, devo ritrasmettere, **ma quando?**
 - Tutti i nodi coinvolti in una collisione (almeno 2) devono ritrasmettere → possibilmente non di nuovo insieme!
 - Devono aspettare **tempi diversi** per la **ritrasmissione** (altrimenti ri-collidono)
- ❖ Strategia di **scelta del tempo di attesa**: funzione del traffico
 - Basso traffico → tempi d'attesa brevi → max. velocità
 - Traffico intenso → tempi d'attesa ben distribuiti, altrimenti ripeto la collisione
- ❖ **Ethernet: Binary Exponential Backoff**: algoritmo di calcolo del tempo di attesa prima della ritrasmissione dopo collisione
 - Si adatta automaticamente alle condizioni di traffico.



Binary Exponential Back-off

❖ **Binary Exponential Back-off Algorithm:**

Se ho collisione

→ invio un **collision burst**

→ Ritrasmetto dopo un **n. casuale di time slots (2τ)** tra 0 e 1

✦ 0 → ritrasmetto subito; 1 → ritrasmetto dopo 2τ

Se ho ancora collisione

→ invio un collision burst

→ Ritrasmetto dopo un **n. casuale di time slots (2τ)** tra 0 e 3

Se ho ancora collisione

→ invio un collision burst

→ ritrasmetto dopo un **n. casuale di time slots (2τ)** tra 0 e 7

...

Se ho ancora collisione

→ invio un collision burst

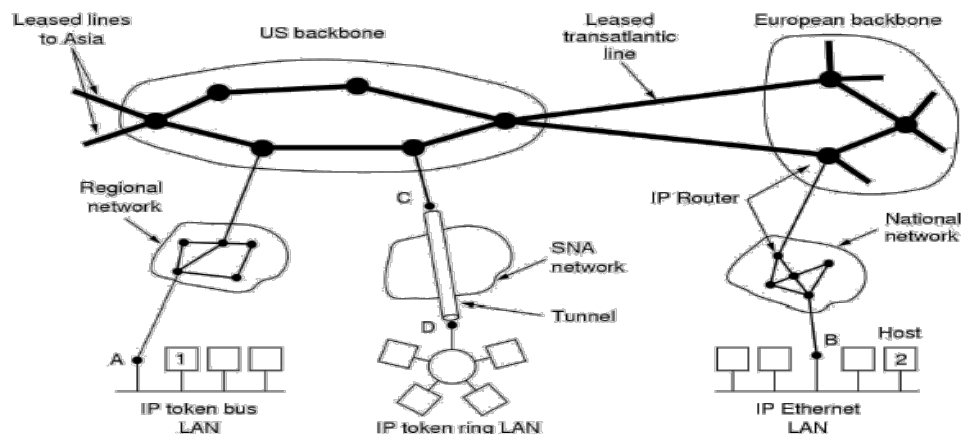
→ ritrasmetto dopo un **n. casuale di time slots (2τ)** tra 0 e 1023

✦ 2 stazioni hanno prob. $1/1024$ di “azzeccare” lo stesso ritardo e collidere

Dopo **16 volte** che ho avuto collisione con N tra 0 e 1023 → **Transmission FAILED**

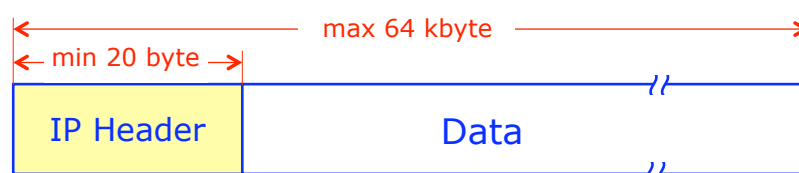
Liv. 3: il livello di rete

- ❖ **Funzioni** del livello di rete
 - **Internetworking:**
Permette collegamento end-to-end, attraverso reti differenti.
 - **Instradamento (routing):**
gestisce la determinazione del percorso da mittente a destinatario (route)
- ❖ L'anima di Internet: **IP (Internet Protocol)**



IP: Internet Protocol

- ❖ **IP:** Protocollo di comunicazione a commutazione di pacchetto – datagramma
 - Protocollo supportato da quasi tutte le reti:
 - ✦ Ethernet, Novell, Token ring, SNA, reti di accesso (PPP), ATM, SDH, Appletalk, ...
- ❖ **Pacchetto di datagramma IP:**
 - dimensione massima: 64 kB
 - dimensione tipica: 1500 bytes → campo data frame Ethernet

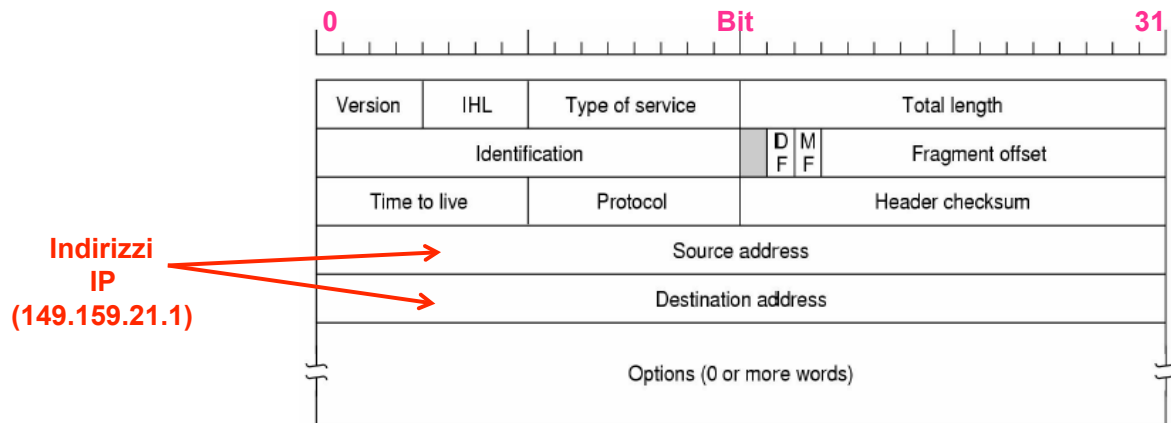


Datagramma IP



IP Header (20+ byte)

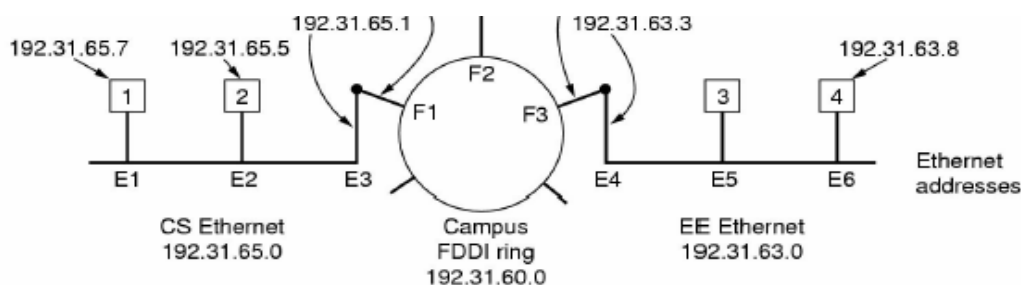
- ❖ **Version:** IPv4 (Internet 2: IPv6)
- ❖ **IHL:** IP header length
- ❖ **Total length:** lunghezza del datagramma completo
- ❖ **ID:** n. identificazione del datagramma
- ❖ **DF, MF, Fragment OFFSET:** informazioni frammentazione del datagramma
- ❖ **Time to live:** timeout di ogni pacchetto (instradamenti errati, ...)
- ❖ **Protocol:** il protocollo di trasporto utilizzato: TCP / UDP
- ❖ **Header Checksum:** calcolato solo sull'header



Pacchetti di controllo



- ❖ **ICMP - Internet Control Message Protocol**
 - messaggi di controllo, per gestire il corretto funzionamento della rete
 - **Diagnostici** (*destination unreachable, Time exceeded, ...*)
 - **Controllo di flusso** (*Source quench, ...*)
 - **Monitoraggio di nodi** *ECHO request/reply (usato da: PING)*
Timestamp request/reply
- ❖ **ARP - Address Resolution Protocol**
 - Collegamento tra **Ethernet** e **IP**
 - **Router:** Pacchetto ARP broadcast: "Chi possiede **IP = x.y.w.z**?"
 - **x.y.w.z:** "Io, ed il mio MAC address è: **a.b.c.d.e.f!**"
 - Il Router può inviare i pacchetti IP con frames Ethernet a **x.y.w.z**





Livello di trasporto (TCP/IP: liv. 3, ISO/OSI: liv. 4)

- ❖ E' il livello che si interfaccia con le applicazioni
 - fornisce i servizi di trasporto informazioni in rete

Funzioni svolte:

- ❖ Fornire un servizio di trasporto affidabile
 - IP non garantisce/controlla l'effettiva consegna dei pacchetti
 - Fornitura di un collegamento connection-oriented
- ❖ Multiplexing: gestione collegamenti multipli dallo stesso nodo IP
 - FTP client e browser web funzionanti in contemporanea necessitano di 2 canali di comunicazione indipendenti, per la stessa coppia sorgente/destinazione

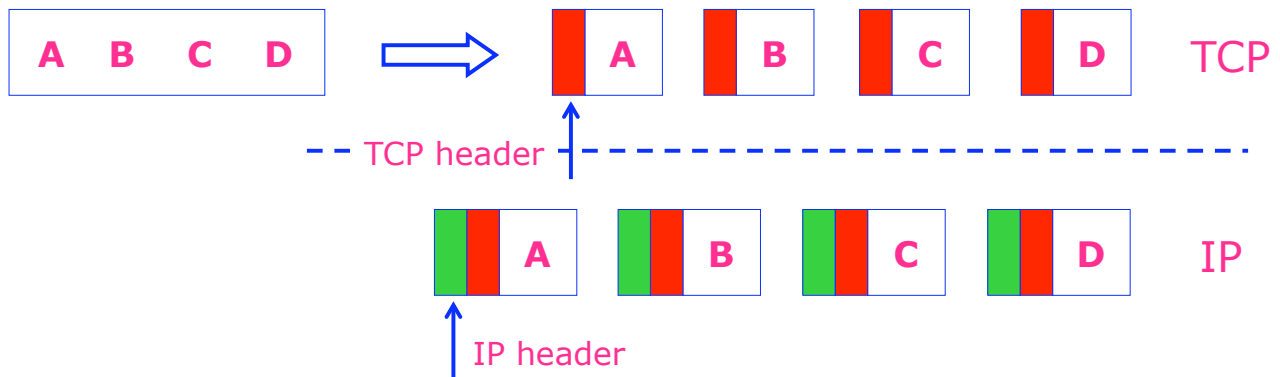


- ❖ **TCP/IP offre 2 protocolli di trasporto:**
- ❖ **TCP - Transmission Control Protocol**
 - Protocollo connection-oriented (utilizzando datagrammi IP)
 - Comunicazione punto-punto, bidirezionale
 - Gestisce il **MULTIPLEXING**: connessioni multiple fra gli stessi 2 indirizzi IP
 - ✦ es: POP, HTTP, FTP, ... tra client e server
 - Canale Affidabile: controlla l'integrità e la completezza della trasmissione
 - Controllo di flusso: regola la velocità di trasferimento in base alle esigenze degli interlocutori
- ❖ **UDP - User Datagram Protocol**
 - Protocollo Datagramma
 - ✦ Di fatto non aggiunge niente al protocollo IP



❖ TCP:

1. Frammenta l'informazione da trasferire e...
2. spedisce ciascun frammento mediante datagramma IP

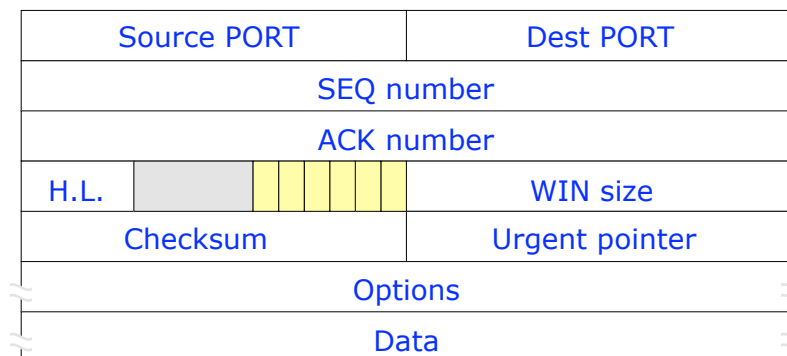


TCP Header



TCP Frame:

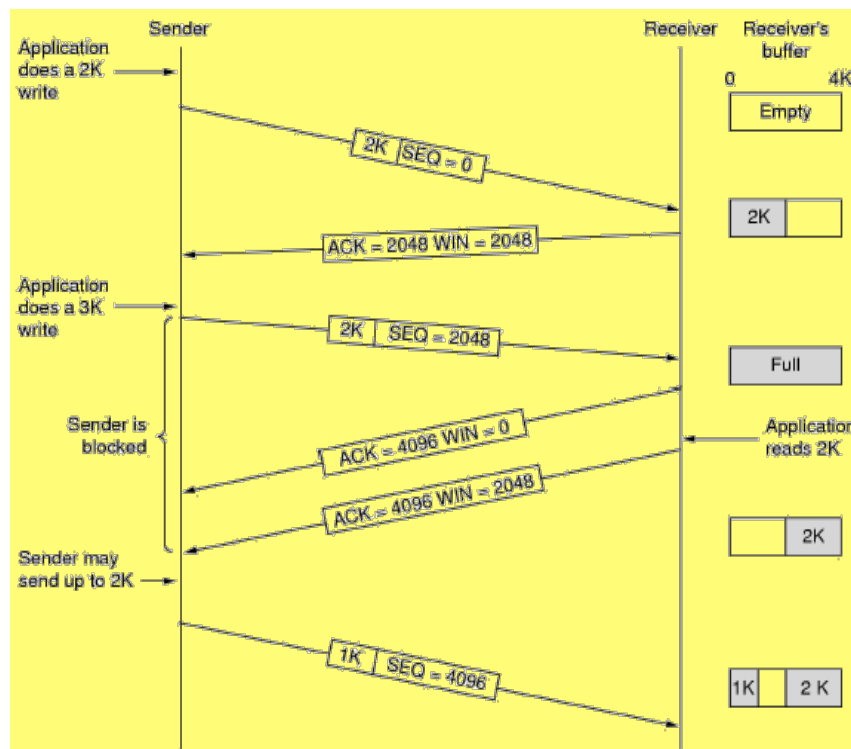
- ❖ **PORT:** definisce il canale logico di comunicazione
- ❖ **SEQ n., ACK n.:** indicano la posizione nel datagramma
- ❖ **URGent, ACKnowledge, PuSH, ReSeT, SYN, FIN:** 6 bit di controllo
- ❖ **WIN SIZE:** dimensione buffer disponibile - quanti byte posso inviare





SERVIZI forniti da TCP:

- ❖ **Controllo integrità della trasmissione (SEQ, ACK)**
 - controllo se sono arrivati **TUTTI** i frammenti
- ❖ **Controllo di flusso efficiente (WIN)**
 - il mittente sa quando deve fermarsi



Livello applicazioni - Internet



- ❖ **Funzione livello applicazione**
 - Fornire all'utente servizi di comunicazione e trasporto informazioni
 - Gestione "trasparente" della rete.
- ❖ Si appoggiano su **UDP**:
 - **DNS** - gestione gerarchica dei domini (www.unimi.it)
 - Utilities minori (Network Time Protocol, messaging (**talk**), ...)
- ❖ Si appoggiano su **TCP**:
 - Collegamento a nodo remoto: **telnet** port=23
 - Scambio files off-line: **ftp** port=20, 21
 - E-mail: **smtp, pop/imap** port=25, 110/143
 - WWW: **http** port=80