



# Reti di calcolatori

*Proff. A. Borghese, F. Pedersini*

Dipartimento di Scienze dell'Informazione  
Università degli Studi di Milano

## Sommario



**Introduzione**

Modelli a livelli

I livelli

## Introduzione

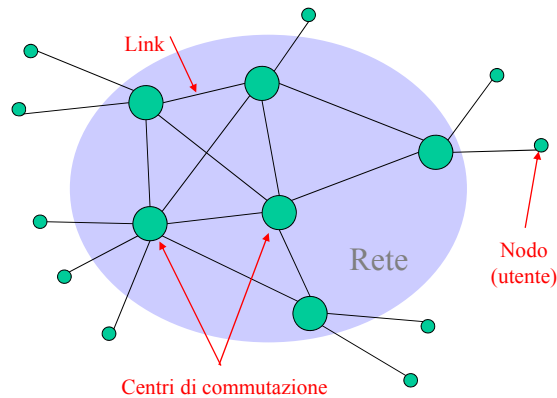


### ❖ Definizione (Tanenbaum: “Computer Network”)

- Rete di calcolatori = insieme di nodi di calcolo indipendenti e interconnessi
  - Es. rete BANCORMAT non è una “computer network”

### ❖ Esempi

- LAN
- Rete wireless
- Internet



## Tassonomia delle reti



- ❖ Non esiste una tassonomia universalmente accettata delle reti.
- ❖ Per **dimensione**
  - bus, LAN, MAN, WAN, Internet
- ❖ Per **topologia**
  - bus, ring, a stella,
- ❖ Per modalità di **collegamento**
  - broadcast, peer-to-peer, ...
- ❖ Per modalità di **traffico**
  - commutazione di circuito, commutazione di pacchetto
- ❖ Per **velocità**
  - 0,001Mbyte/s a 1,000Mbyte/s

## Tassonomia – dimensioni



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

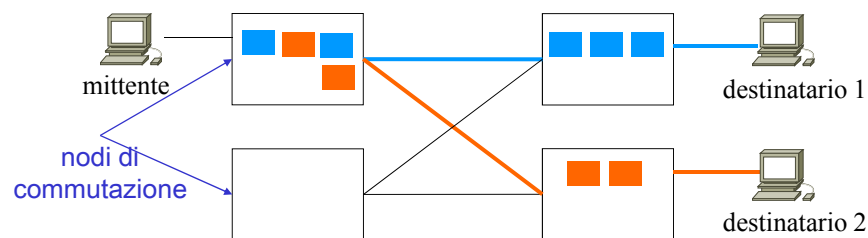
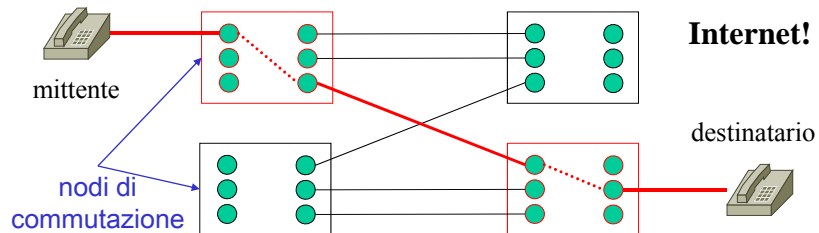
Local Area Network  
(LAN)

Metropolitan Area Network  
(MAN)

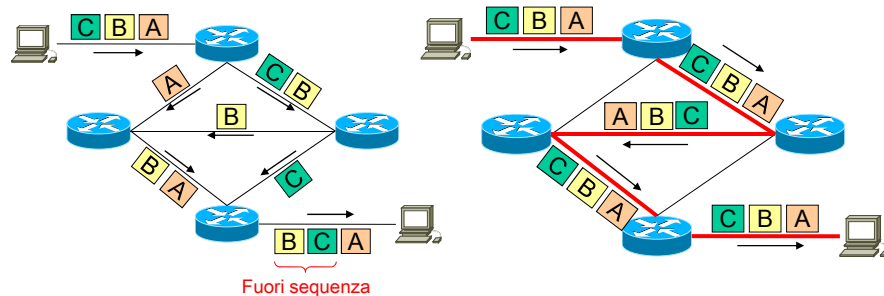
Wide Area Network (WAN)

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.

## Aspetti generali di funzionamento



- ❖ Esempio: **FTP su LAN a bus (Ethernet)**
  - Collegamento fisico tra i terminali (PC)
    - mezzo trasmissivo (MT)
  - Meccanismo di comunicazione dati
    - Tecnica di **trasmissione digitale** sfruttando il MT
  - Protocollo di dialogo tra i vari terminali
    - Chi parla? Quando? Chi ascolta? quando?
  - 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 SW sul terminale utente
- ❖ Perché la rete funzioni devo progettare/gestire/sincronizzare tutti questi aspetti → sistema complesso

# Sommario



## Introduzione

### Modelli a livelli

### I livelli

# Modelli di rete a livelli



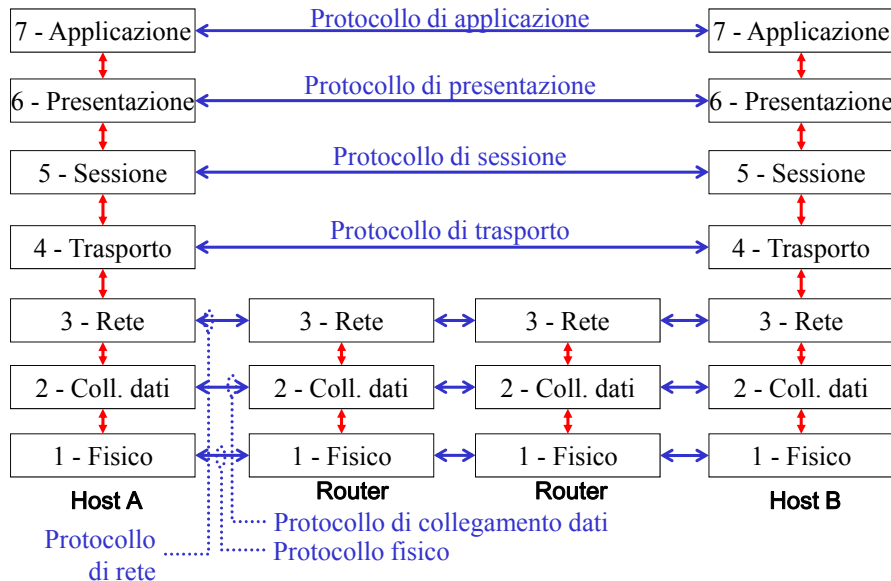
- ❖ Soluzione:
  - Standardizzazione (che prevede **tolleranza agli errori** di trasmissione)
  - Dividere il problema in sottoproblemi → Modello di rete a **LIVELLI**
- ❖ Definisco singolarmente ogni livello:
  - definisco il suo funzionamento → **Protocollo**
  - definisco il suo modo di interagire con i livelli adiacenti (sopra e sotto) → **Interfaccia**
- ❖ Modelli **standard di rete**:
  - **TCP-IP** (4 livelli)
  - **ISO-OSI** (7 livelli)

# Modello ISO-OSI: i 7 livelli



Numero di layer	Nome del layer	Funzione principale	Esempio protocollo	Componente della rete
7	Application	Gestisce la trasmissione a livello di applicazione SW	FTP, DNS, NFS, http	Gateway, smart switch
6	Presentation	Si preoccupa della coerenza sintattica dell'informazione presentata al livello 7 Traduce da applicazione al formato della rete e viceversa	-	Gateway
5	Session	Gestisce ogni singolo "dialogo" sulla rete, instaurando il collegamento, negoziando le regole di comunicazione e chiudendo il collegamento.	-	Gateway
4	Transport	Servizio <i>affidabile</i> di trasferimento dell'informazione per il livello superiore	TCP	Gateway
3	Network	Instradamento dei pacchetti. Traduce gli indirizzi logici in indirizzi fisici (e.g. il nome di un computer nel suo indirizzo MAC)	IP	Router, ATM switch
2	Data Link	Traduce i pacchetti in sequenze di bit ed assembla le sequenze di bit in pacchetti	IEEE802	Bridge, Network, Interface card
1	Physical	Definizione del mezzo trasmissivo ( <i>MT</i> ). Trasmette stream di bit verso il cavo di rete.	IEEE802	Hub

# Lo stack ISO-OSI



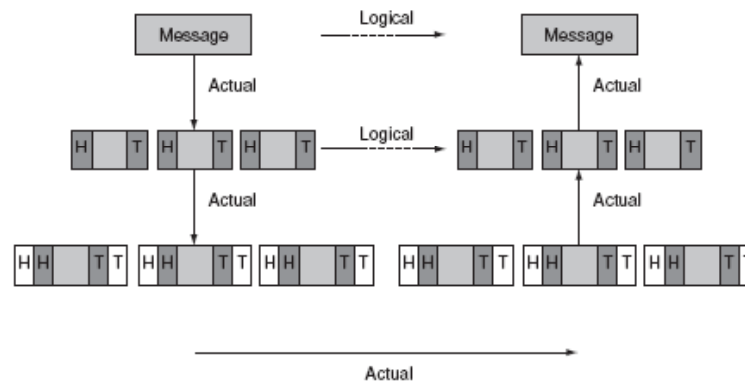
# Pacchettizzazione



Il contenuto di un messaggio viene suddiviso in pacchetti.

Ciascun pacchetto è auto-contenuto e contiene indirizzo, e contenuto.

Il messaggio viene riassembleato a lato receiver.



A.A. 20C

13/37

# Modello TCP-IP: i 4 livelli



TCP/IP definisce un formato standard per i vari pacchetti.

## 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

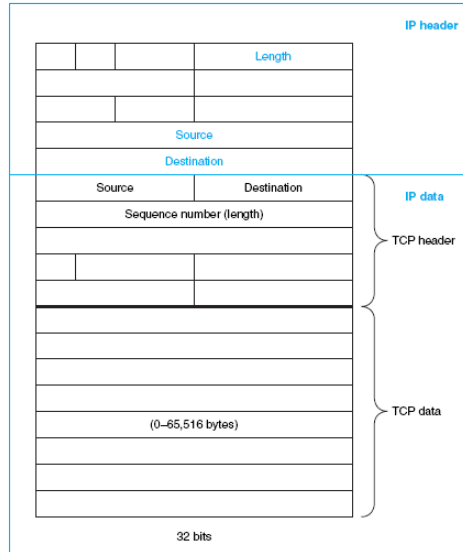
- Condensa i livelli 5-7 ISO-OSI
- telnet, FTP, e-mail, www

A.A. 2009/10

Copyright: A. Borghese, F. Pedersini – DSI, UniMI

14/37

## Il formato TCP-IP



**TCP/IP**- Transmission Control Protocol/Internet Protocol

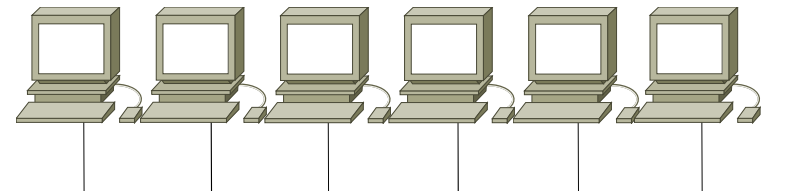
**IP** - is responsible for moving packet of data from node to node. It operates on gateway machines that move data from department to organization to region and then around the world.

**TCP** - is responsible for verifying the correct delivery of data from client to server.

## Case study: TCP/IP su rete locale (LAN)

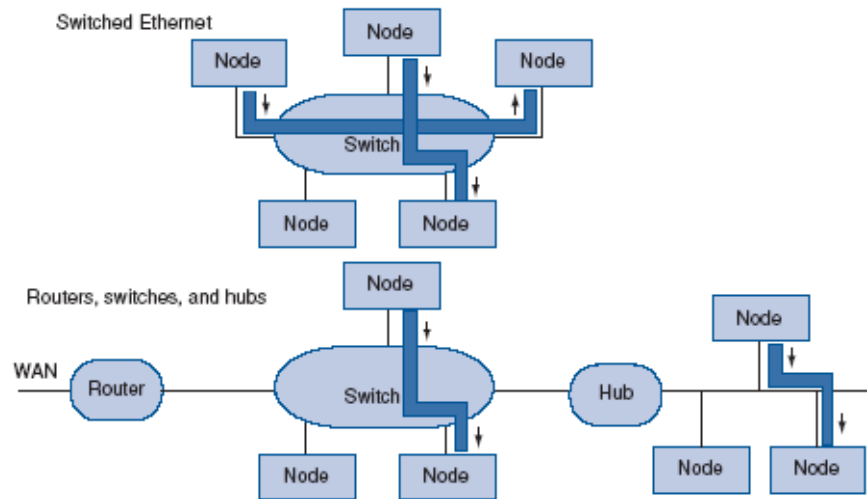


- ❖ Caso più diffuso di rete locale
  - LAN di Windows, UNIX/LINUX, ...
  - Topologia: bus, mezzo trasmissivo: cavo in rame
  - Comunicazione: **Ethernet** (IEEE 802.3)
  - Protocolli **IP, TCP** (Internet)





# I dispositivi principali



# Sommario



Introduzione

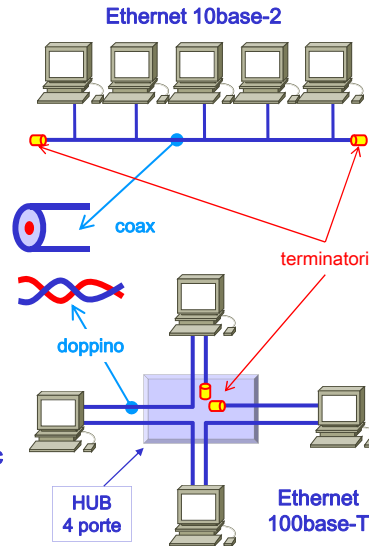
Modelli a livelli

**I livelli**

## Il livello fisico in Ethernet (IEEE 802.3)



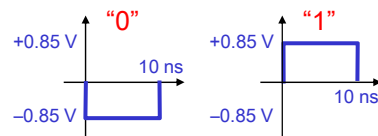
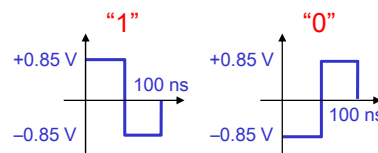
- ❖ Necessario un terminatore per simulare lunghezza infinita
- ❖ Diversi standard:
  - Cavo **coassiale**, bus terminato (10base-2, 10base-5)
    - Capacità di canale: **10 Mb/sec**
    - $L_{MAX} = 2500$  m
  - Cavo a **doppino** (100base-T)
    - Doppino, pseudo-stella (bus)
    - Capacità di canale: **100 Mb/sec**
    - $L_{MAX} = 100$  m
  - Fibra ottica (10,100base-F)
  - Gigabit Ethernet (1000Base-T)
    - Capacità di canale: **fino a 40 Gb/sec**
    - $L_{MAX} = 100$  m



## Livello fisico in Ethernet: codifica



- ❖ Tecnica di trasmissione
  - come viaggia l'informazione binaria sul cavo?
- ❖ **10 Mb/sec:**
  - Manchester Encoding**
    - autosincronizzante
    - robusto → circuiti semplici
    - ridondanza: 100 %
- ❖ **100 Mb/sec:**
  - 4B/5B – aggiunta 5° bit sync**
    - meno robusto
    - → circuiti + sofisticati
    - ridondanza: 25 %



Sequenza: 101000000101..

## Livello di collegamento dati



### ❖ Funzioni:

- **trasmissione affidabile sul MT**
  - Sincronizzazione affidabile → **framing**
  - Controllo errori → **codici controllo errore**
  
- **Gestione del MT condiviso** → politica di arbitraggio
  - **MAC: Medium Access Control**
  
- **Framing:**
  - trasmetto informazione in pacchetti di lunghezza limitata
  
- **Controllo errori:**
  - allego ad ogni frame informazioni per rivelare errori di trasmissione

## MAC: protocolli statici e dinamici



### ❖ 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 (WiFi)

### ❖ Dinamici

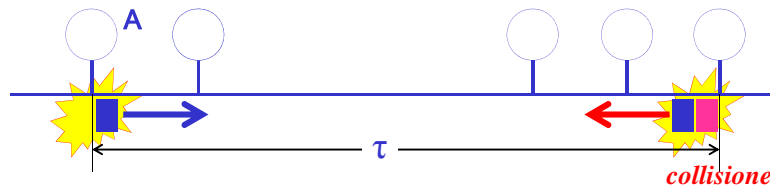
- **ALOHA**
  - ogni utente inizia a trasmettere quando vuole
  - se due o più trasmissioni collidono vengono distrutte
  - attraverso un meccanismo di feedback gli utenti si accorgono delle collisioni e ritentano la trasmissione dopo un tempo casuale
- **“Carrier Sense Multiple Access” (CSMA)**
  - Prima di trasmettere ogni utente “ascolta” se la linea è libera.
  - Periodo di vulnerabilità ridotto al tempo di propagazione (cf. clock skew).
- **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



### ❖ CSMA/CD:

- Prima di trasmettere, ogni utente "ascolta" se la linea è libera
- Se sì, l'utente inizia la trasmissione, ma resta in ascolto per rivelare eventuali collisioni
- Se 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  $2\tau$
- Per  $D_{MAX} = 2,5 \text{ km}$ ,  $r = 10 \text{ Mb/s}$  →  $2\tau = 51,2 \text{ } \mu\text{sec} = 512 \text{ bit} = 64 \text{ bytes}$
- Per far sapere a tutti della collisione, emetto un collision burst (durata  $> 2\tau$ )



## Gestione delle collisioni – ritrasmissione



- ❖ Se ho collisione, devo ritrasmettere
  - Tutti i nodi coinvolti in una collisione (almeno 2) devono ritrasmettere → possibilmente non di nuovo insieme!
  - Devo aspettare tempi diversi per la ritrasmissione
- ❖ Strategia di scelta del tempo di attesa: funzione del traffico
  - Basso traffico → tempi di 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



### ❖ ALGORITMO

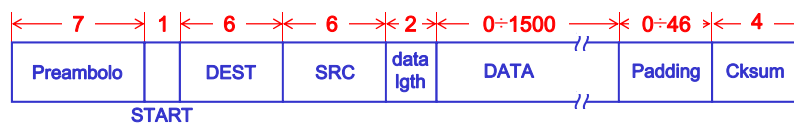
- Se ho collisione → invio un collision burst
- Ritrasmetto dopo un n. casuale di time slots (2τ) tra 0 e 1
  - 0 → ritrasmetto subito; 1 → ritrasmetto dopo 2τ
- Se ho collisione → invio un collision burst
- Ritrasmetto dopo un n. casuale di time slots (2τ) tra 0 e 3
- Se ho collisione → invio un collision burst → ritrasmetto dopo un n. casuale di time slots (2τ) tra 0 e 7
- ...
- Se ho 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 collisione con N tra 0 e 1023 → **FAILED**

## Il frame Ethernet (802.3)



### ❖ “Pacchetto” di lunghezza variabile: 64 ÷ 1528 byte (MAX)

- **Preambolo**: 7 bytes: 10101010 di sincronizzazione
- **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 ÷ 1500)
- **DATA** (0 ÷ 1500 bytes)
- **PADDING** – (0 ÷ 46 bytes) necessario per raggiungere la lunghezza minima di frame di 64 bytes
- **CHECKSUM** – (4 bytes) codice rivelatore di errori

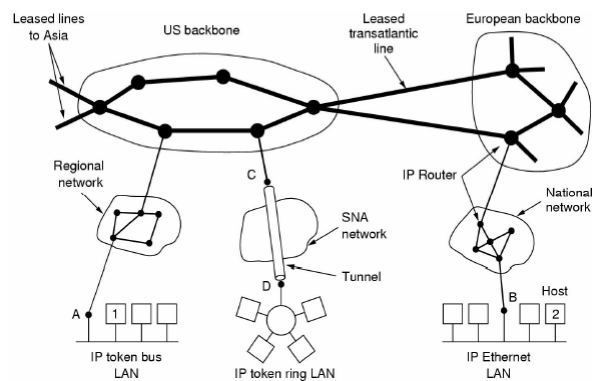


## Il livello di rete



- ❖ Funzioni
  - Permette collegamento end-to-end, attraverso reti differenti  
→ **Internetworking**
  - **Instradamento (routing)**: gestisce la determinazione del percorso da mittente a destinatario (route)

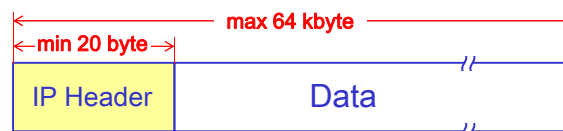
- ❖ L'anima di Internet:  
**IP (Internet Protocol)**



## IP



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

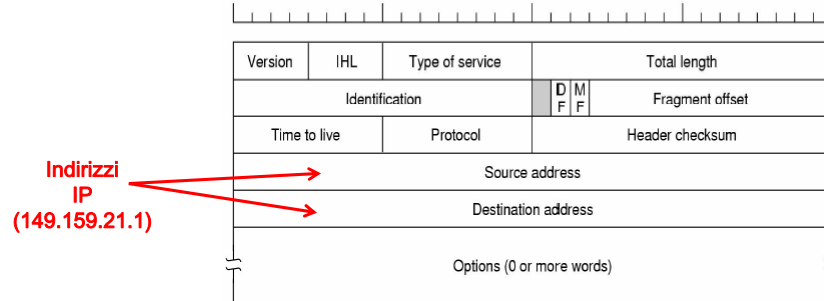


Pacchetto datagramma IP

## IP Header (20+ byte)



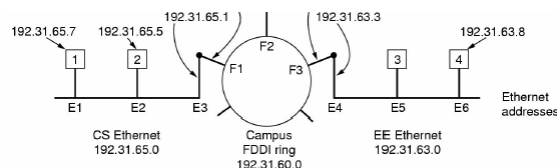
- ❖ **Version:** IPv4 (Internet 2: IPv6)
- ❖ **IHL:** header length
- ❖ **Total length:** lunghezza del datagramma completo
- ❖ **ID:** n. identificazione del datagramma
- ❖ **DF, MF, Fragment OFFSET:** informazioni frammentazione del datagramma
- ❖ **Time to live:** timeout di ogni pacchetto (instradamenti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



## Pacchetti di controllo



- ❖ **ICMP – Internet Control Message Protocol**
  - messaggi per controllare il 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



- ❖ E' il livello che si interfaccia con le applicazioni
  - Fornisce una comunicazione logica tra processi.
  - Fornisce i servizi di trasporto informazioni in rete.
- ❖ Funzioni
  - 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 → 2 canali di comunicazione indipendenti

## Livello di Trasporto in Internet: TCP, UDP



- ❖ 2 servizi offerti:
- ❖ **TCP – Transmission Control Protocol**
  - Protocollo connection-oriented (utilizzando datagrammi IP)
  - Comunicazione punto-punto, bidirezionale
  - 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

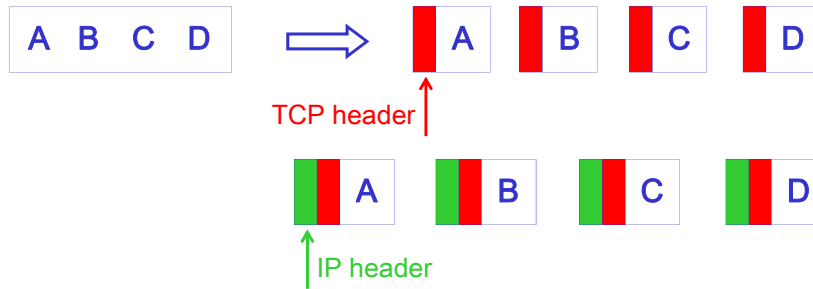
EEC in the header



# TCP



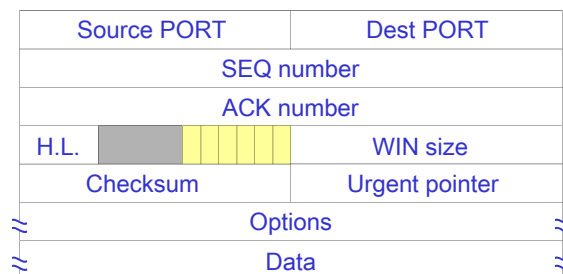
- ❖ TCP prende l'informazione da trasferire, la frammenta e la spedisce mediante datagrammi IP:



# TCP Header



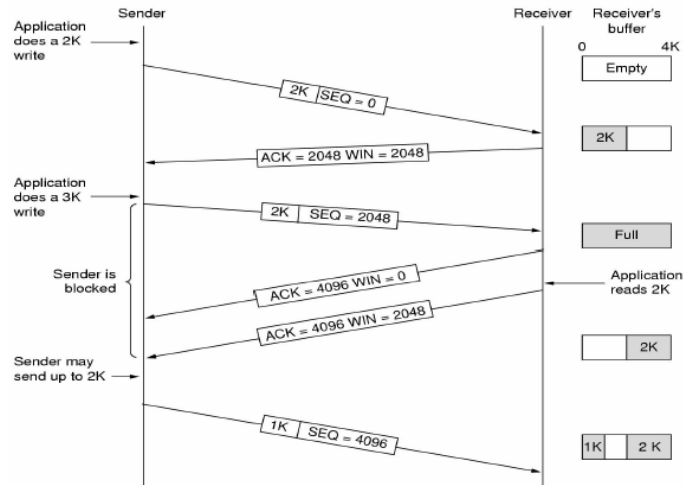
- ❖ **PORT**: definisce il canale logico di comunicazione
- ❖ **SEQ, ACK**: indicano la posizione nel datagramma
- ❖ **URGent, ACKnowledge, PuSH, ReSeT, SYN, FIN**
- ❖ **WIN SIZE**: dimensione buffer disponibile – quanti byte posso inviare



## TCP: protocollo comunicazione



- ❖ Controllo integrità della trasmissione
  - controllo se sono arrivati TUTTI i frammenti
- ❖ Controllo di flusso efficiente
  - il mittente sa quando deve fermarsi



A.A. 2009/10

Copyright: A. Borghese, F. Pedersini – DSI, UniMI

35/37

## Livello applicazioni – Internet



- ❖ Funzione
  - Fornire all'utente servizi di comunicazione e trasporto informazioni. Operano mediante [socket](#).
  - Gestione "trasparente" della rete.
- ❖ Si appoggiano su **UDP**:
  - **DNS** – gestione gerarchica dei [domini](#) ([www.unimi.it](http://www.unimi.it))
  - Utilities minori (Time Protocol, messaging ([talk](#)), ...)
  - Voice over IP (VoIP), Streaming Video
- ❖ Si appoggiano su **TCP**:
  - Collegamento a nodo remoto: **telnet** port=23
  - Scambio files off-line: **ftp** port=20,21
  - Scambio files on-line: **nfs**
  - E-mail: **smtp, pop/imap** port=25,110/143
  - **WWW** port=80

A.A. 2009/10

Copyright: A. Borghese, F. Pedersini – DSI, UniMI

36/37

# Sommario

---



Introduzione

Modelli a livelli

I livelli