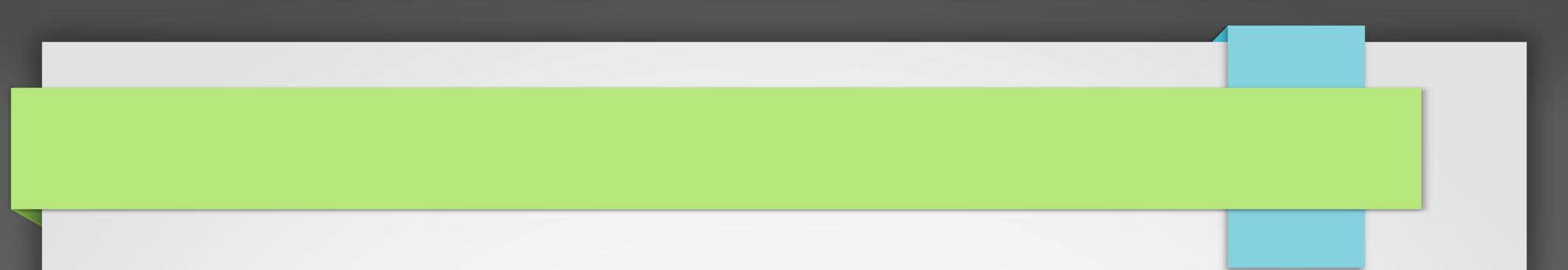


Introduzione all'Informatica

Loriano Storchi

loriano@storchi.org

<http://www.storchi.org/>



UNITA' DI MISURA E PRESTAZIONI

Prestazioni

- **Bandwidth** (Larghezza di banda) la quantità di dati (numero di bit) massima di dati che possono essere trasmessi in un canale, La si usa **spesso come approssimazione del rendimento effettivo** (unita' di misura ad esempio **Mbps**)
- **Throughput (Rendimento)** quantità di dati (numero di bit) trasmessi sul canale in un certo periodo di tempo (unita' di misura ad esempio **Mbps**)

Prestazioni

- **Latenza (latency)** o anche ritardo (**delay**) e' il tempo impiegato da un messaggio per andare da un punto all'altro (**unita' di misura il tempo ad esempio ms = millisecondo = (1/1000) s**)
 - Tempi di trasmissione dovuti alla **banale velocita' di propagazione del segnale nel mezzo trasmissivo**, e dunque alla **distanza**
 - **Tempi richiesti per elaborare ad esempio l'intestazione dei pacchetti trasmessi**
- **Round Trip Time (RTT)** tempo impiegato dal messaggio per andare da un punto A ad un punto B e tornare nuovamente da B ad A (**ping e traceroute**)

Prestazioni

```
redo@eeegw:~$ traceroute www.google.com
traceroute to www.google.com (216.58.205.36), 30 hops max, 60 byte packets
 1 gw.ego.eco (192.168.10.1)  0.497 ms  0.410 ms  0.369 ms
 2 maingw.ego.eco.168.192.in-addr.arpa (192.168.1.1)  1.395 ms  1.278 ms  1.621 ms
 3 * * *
 4 172.18.25.226 (172.18.25.226)  10.313 ms 172.18.25.208 (172.18.25.208)  10.438 ms 172.18.25.230 (172.18.25.230)  10.527
 5 172.18.24.73 (172.18.24.73)  13.664 ms 172.18.24.85 (172.18.24.85)  13.472 ms 172.18.24.81 (172.18.24.81)  12.449 ms
 6 172.19.240.189 (172.19.240.189)  24.534 ms 20.669 ms 20.618 ms
 7 * * *
 8 74.125.51.148 (74.125.51.148)  29.704 ms 72.14.219.236 (72.14.219.236)  35.951 ms 74.125.48.192 (74.125.48.192)  15.297
 9 * * *
10 216.239.42.31 (216.239.42.31)  14.514 ms  14.070 ms  17.031 ms
11 mil04s24-in-f36.1e100.net (216.58.205.36)  15.526 ms  26.030 ms  16.210 ms
```

```
[redo@banquo ~]$ ping 10.0.63.254
PING 10.0.63.254 (10.0.63.254) 56(84) bytes of data.
64 bytes from 10.0.63.254: icmp_seq=1 ttl=64 time=0.209 ms
64 bytes from 10.0.63.254: icmp_seq=2 ttl=64 time=0.196 ms
64 bytes from 10.0.63.254: icmp_seq=3 ttl=64 time=0.181 ms
^C
--- 10.0.63.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2074ms
rtt min/avg/max/mdev = 0.181/0.195/0.209/0.016 ms
[redo@banquo ~]$ ping www.google.com
PING www.google.com (216.58.205.196) 56(84) bytes of data.
64 bytes from mil04s29-in-f196.1e100.net (216.58.205.196): icmp_seq=1 ttl=55 time=24.5 ms
64 bytes from mil04s29-in-f196.1e100.net (216.58.205.196): icmp_seq=2 ttl=55 time=24.9 ms
64 bytes from mil04s29-in-f196.1e100.net (216.58.205.196): icmp_seq=3 ttl=55 time=25.0 ms
^C
--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 24.582/24.875/25.095/0.215 ms
```

Prestazioni

```
redo@raspberrypi:~$ curl -s https://raw.githubusercontent.com/sivel/speedtest-cl
i/master/speedtest.py | python -
Retrieving speedtest.net configuration...
Testing from Telecom Italia (79.27.179.48)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by Inweb Adriatico S.r.l. (Silvi) [18.03 km]: 17.92 ms
Testing download speed.....
.....
Download: 35.64 Mbit/s
Testing upload speed.....
.....
Upload: 15.45 Mbit/s
redo@raspberrypi:~$ □
```

Prestazioni

```
root@raspberrypi:~# iperf -s
```

```
-----  
Server listening on TCP port 5001  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 4] local 192.168.10.22 port 5001 connected with 192.168.10.218 port 40518  
[ ID] Interval      Transfer    Bandwidth  
[ 4]  0.0-10.1 sec  113 MBytes  94.2 Mbits/sec
```

```
root@buchner: ~
```

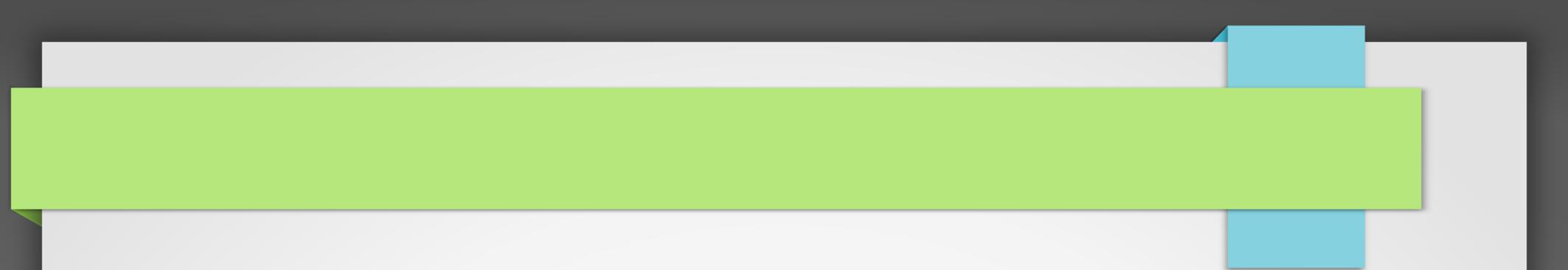
```
root@buchner:~# iperf -c rpi.ego.eco
```

```
-----  
Client connecting to rpi.ego.eco, TCP port 5001  
TCP window size: 85.0 KByte (default)  
-----
```

```
[ 3] local 192.168.10.218 port 40518 connected with 192.168.10.22 port 5001  
[ ID] Interval      Transfer    Bandwidth  
[ 3]  0.0-10.0 sec  113 MBytes  94.6 Mbits/sec  
root@buchner:~#
```

Unita' di misura

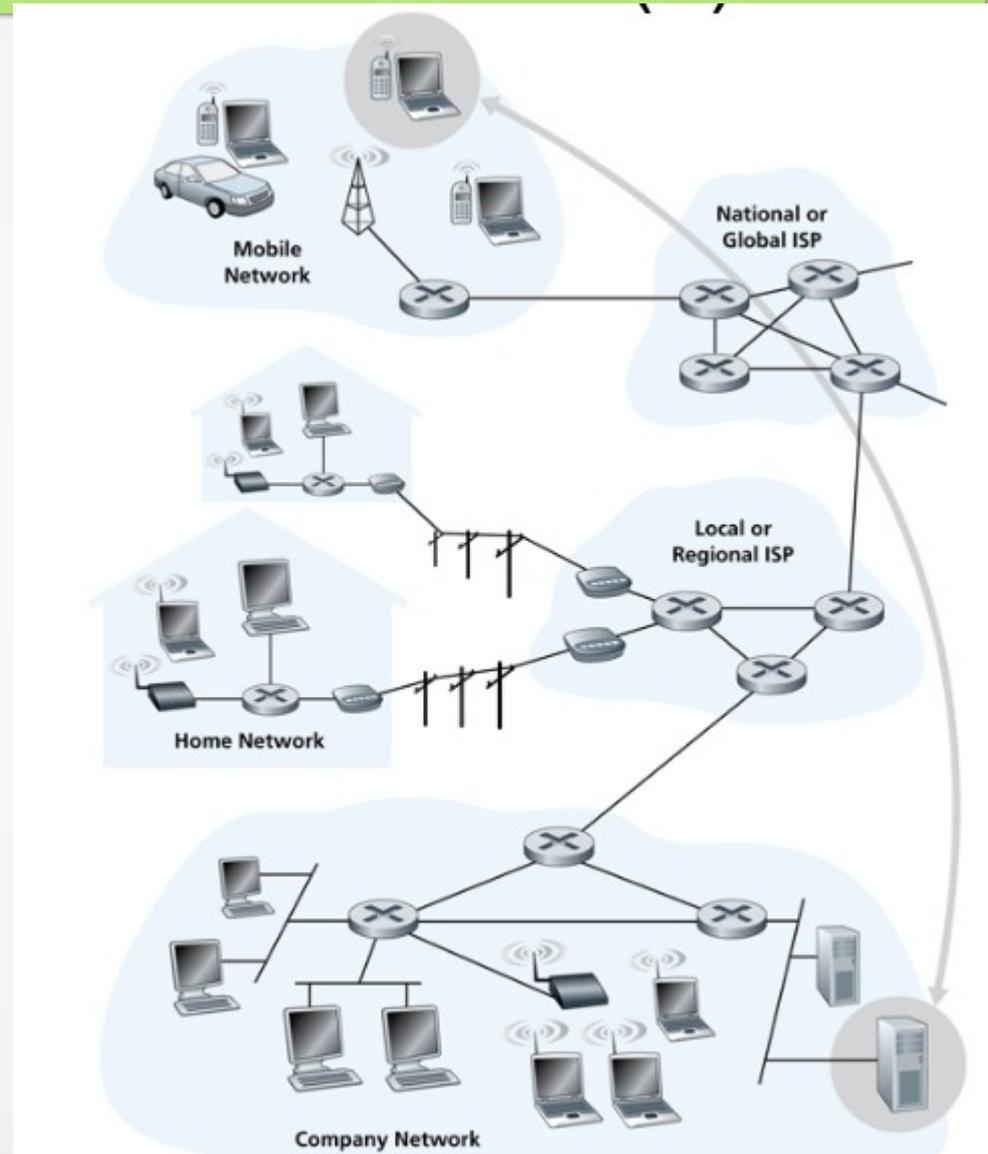
- **Velocita' trasmissione dati = quantita' di informazione / tempo di trasferimento**
- In generale questa velocita' viene espressa in bit per secondo cioe' **bit/s** (oppure **bps** detta anche **bit rate**). Si usa anche il byte per secondo **byte/s** (oppure **Bps**).
- Si usano poi i prefissi standard **k (=kilo 10^3)** , **M (=mega 10^6)**, **G (=giga 10^9)**, quindi non le approssimazioni basate sulle potenze di due che si utilizzano in ambito informatico.
- Convertire da **bps a Bps e' semplice basta dividere per 8**. Ad esempio ADSL **10 Mbps = 10 Mbps/8 = 1250 KBps**
- Dovendo trasferire un file da **10 MiB** con una linea da **5 Mbps** impieghero' circa **$(10 * 1024 * 1024 * 8) / 5 * 10^6 = 16.8$ s** (trascurando la latenza)



RETI DI COMPUTER

Reti di Calcolatori

Un insieme di calcolatori autonomi collegati, la rete e' vista come una **fornitrice di canali logici** attraverso i quali le varie applicazioni possono comunicare fra di loro.

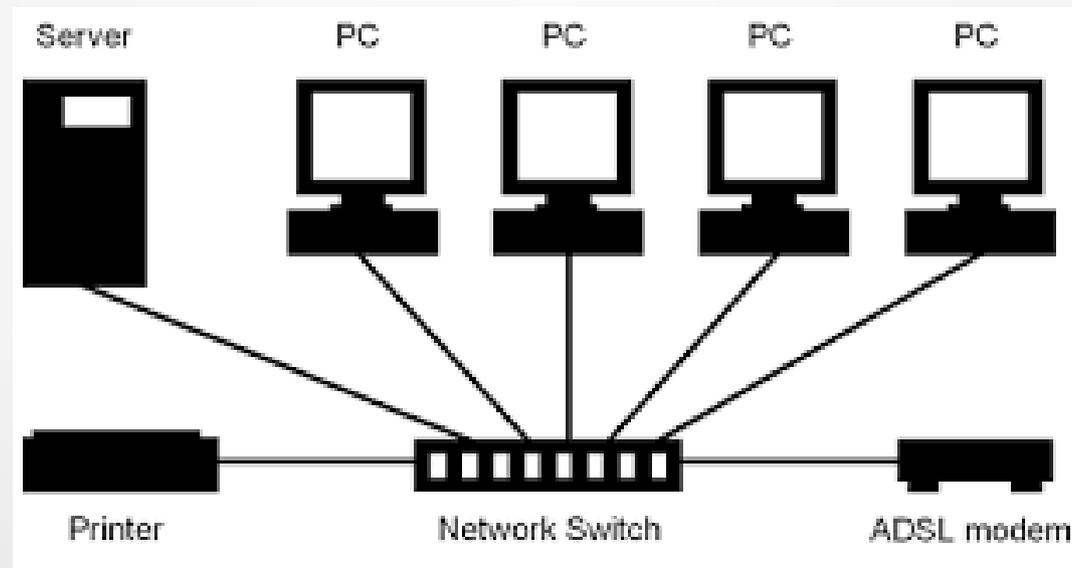


Reti di Computer

- Un insieme di calcolatori autonomi collegati
 - **Nodi (host)** possono essere sia server che desktop PC, oppure dispositivi mobili o altro
 - Collegamenti, sono i canali che permettono ai nodi di comunicare. Ovviamente posso avere **mezzi di trasmissioni molto diversi, ad esempio il doppino telefonico, oppure fibre ottiche, oppure canali radio (wireless)**
 - Posso avere collegamenti diretti od indiretti (**switch**)

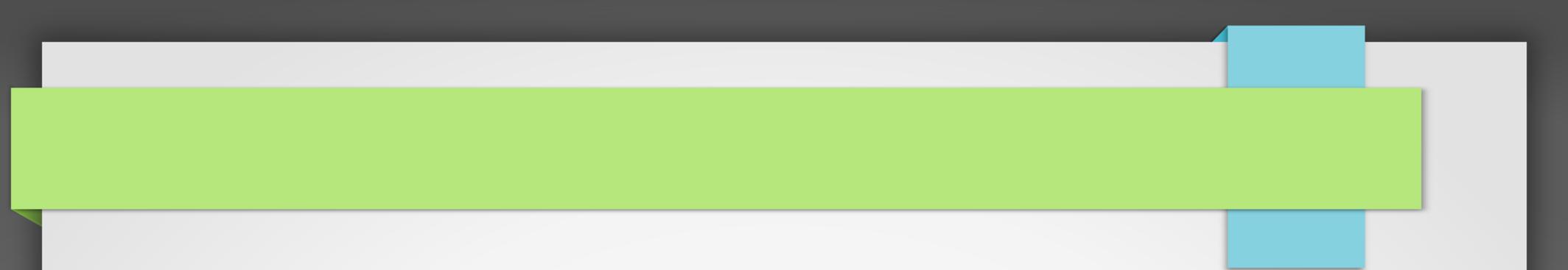
Switch

- switch di rete (chiamato anche hub di commutazione, hub di bridging) è un dispositivo di rete che collega i dispositivi insieme computer utilizzando la **commutazione di pacchetto** per ricevere, elaborare e inoltrare i dati al dispositivo di destinazione.



Reti di Computer

- Posso caratterizzare le reti in base alla loro estensione
 - **LAN (Local Area Network)**: rete su scala locale, si tratta di reti estese a livello di una singola stanza o al massimo di un edificio
 - **MAN (Metropolitan Area Network)**: puo' ad esempio collegare piu' LAN
 - **WAN (Wide Area Network)**: reti estese su aree geografiche. Connettono assieme LAN e MAN (Internet e' la WAN per eccellenza)

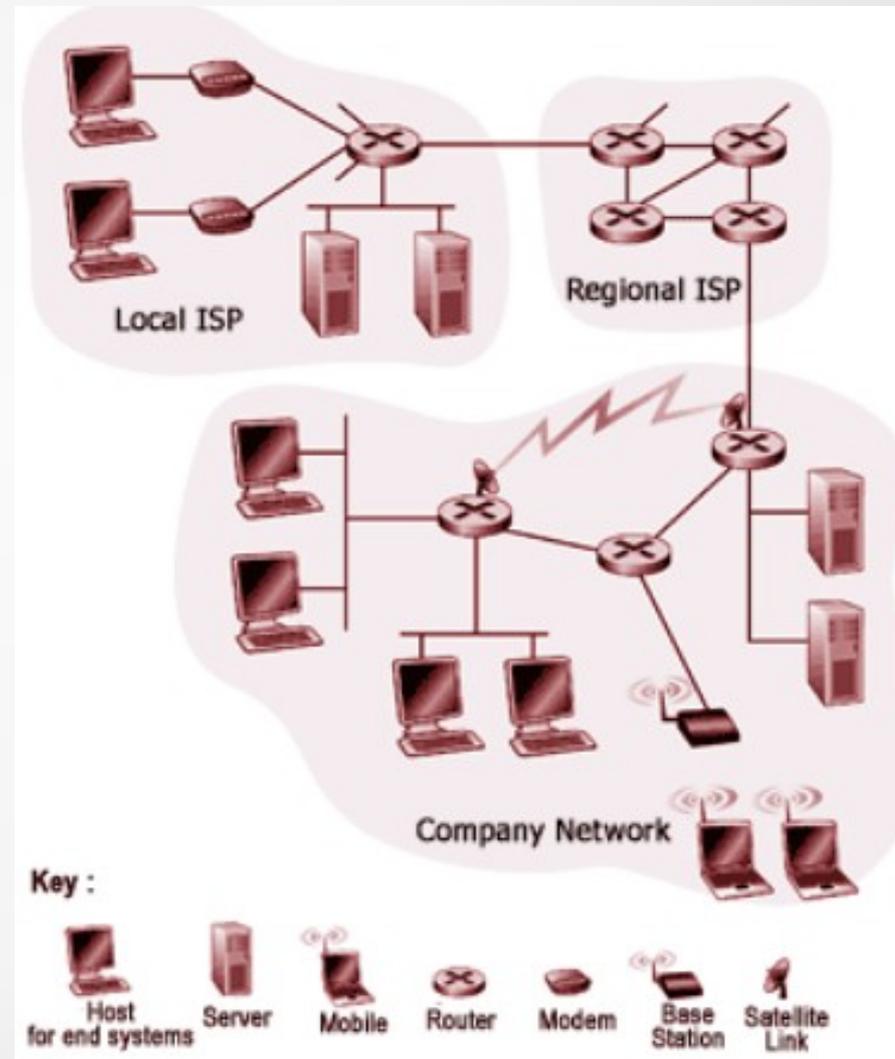


INTERNET

Internet

Possiamo descrivere internet innanzi tutto **dal punto di vista della componentistica di base**.

Internet interconnette milioni di dispositivi in tutto il mondo, i tradizionali PCs desktop, i dispositivi mobili, ed i così detti servers (che immagazzinano e trasmettono dati, come pagine html, e-mail etc etc). Questi dispositivi sono detti **hosts o end systems**

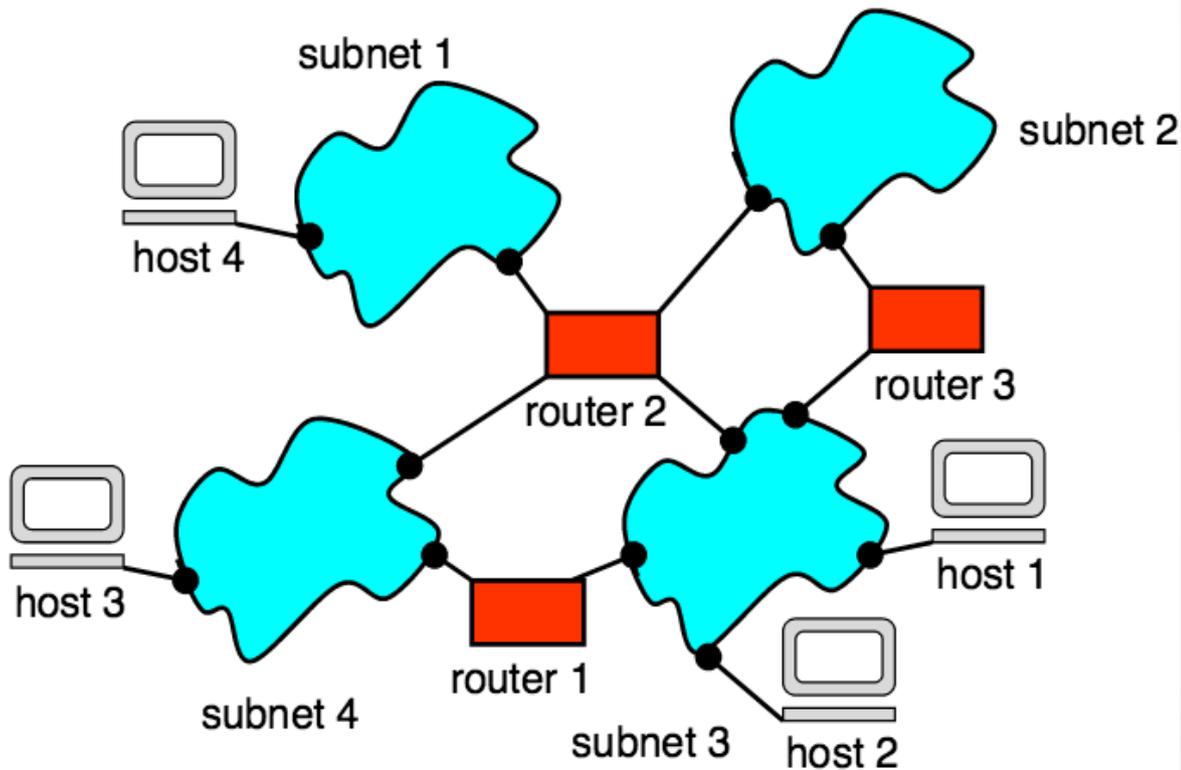


Internet

- I vari hosts sono interconnessi mediante canali di comunicazione. Questi canali possono essere di natura molto diversa , ad esempio cavi coassiali, fili di rame, fibre ottiche o ponti radio
- **Generalmente gli hosts non sono interconnessi direttamente ma ci sono dei dispositivi di “switching” detti router. I router servono ad instradare il traffico dati , e quindi prendono informazioni che arrivano da un canale e le inviano ad un altro canale**
- Internet e' un insieme di reti interconnesse fra di loro . **I vari hosts, cosi' come gli altri devices infrastrutturali comunicano fra di loro usando protocolli comuni (IP Internet Protocol e TCP Transmission Control Protocol sono i due protocolli principali)**

Internet

Internet oggi interconnette migliaia di sottoreti



Host: computer collegato ad internet, puo' essere sia un client che un server a livello applicativo

Router: nodo che serve ad instradare il traffico (**a Layer 3 network gateway device,**)

Sottorete: insieme di host fra cui c'e' un collegamento di livello 2 (ad esempio una LAN)

Ad oggi ordine di miliardi di Host

Internet: Breve storia

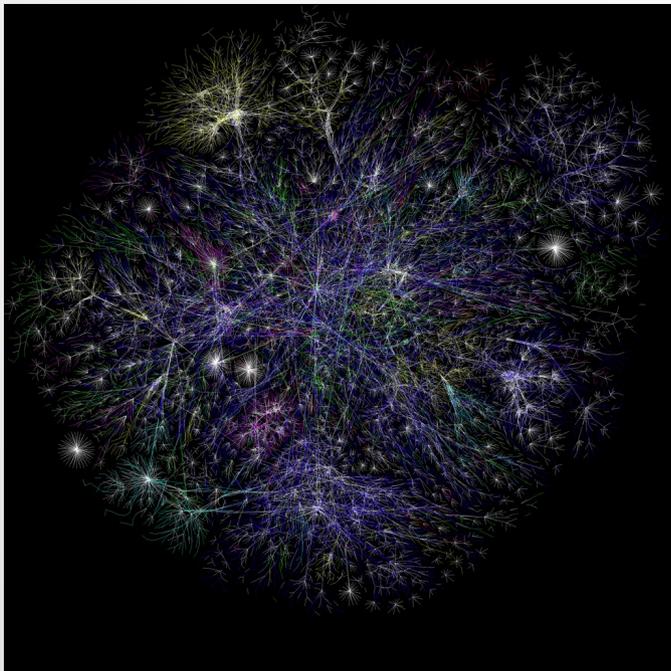
- **1969 parte la progettazione della rete militare ARPANET** . Questa rete viene progettata fra le altre cose anche con lo scopo di resistere ad attacchi nucleari. Infatti era in grado in caso di connettere i dispositivi interconnessi seguendo in caso di rotture strade diverse
- **1972 nasce la posta elettronica (e-mail) trasferimento di files via FTP e collegamento remoto (Remote Login)**
- **1974 vengono ufficialmente presentati i protocollo IP e TCP.**
- **1979 data di nascita della rete CSNet che interconnette le Università' ed i centri di ricerca degli Stati Uniti**

Internet: Breve storia

- **1982 ARPANET e CSNet** vengono collegate, questa viene considerata la **data di nascita ufficiale** in qualche modo di Internet
- **1990 NSFNet, rete che collega supercomputer soppianta Arpanet.** Questo apre la strada ad usi civili e commerciali di Internet.
- **1990** Stesso anno **Tim Bernes-Lee** che allora lavorava al **CERN di Ginevra inventa l'HTML** (Hyper Text Markup Language) che consente la gestione informazioni di natura diversa, testo, immagini etc etc. Questo e' il primo passo verso il **WWW (World Wide Web)**

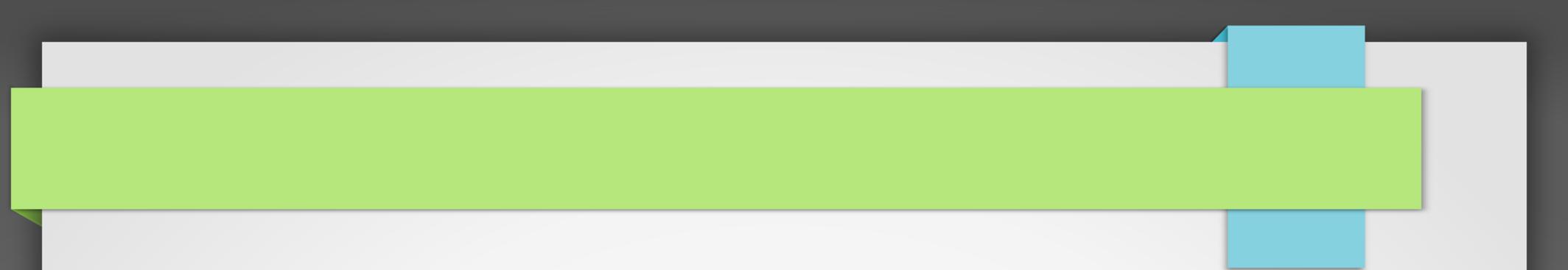
Internet: Breve storia

- **1993** viene realizzato **Mosaic** il primo browser
- **1994 nasco Yahoo!** Il primo motore di ricerca . Nella seconda meta' degli anni 90 ne vengono realizzati molti altri fino ad arrivare al **1998 data della nascita di Google.**



Internet oggi contiene circa **10 miliardi di computer, ogni computer (dispositivo) contiene in media un paio di miliardi di transistor. Quindi internet interconnette 10^{19} transistor , in conclusione ci sono 10000 volte piu' transistor che sinapsi nel cervello umano**

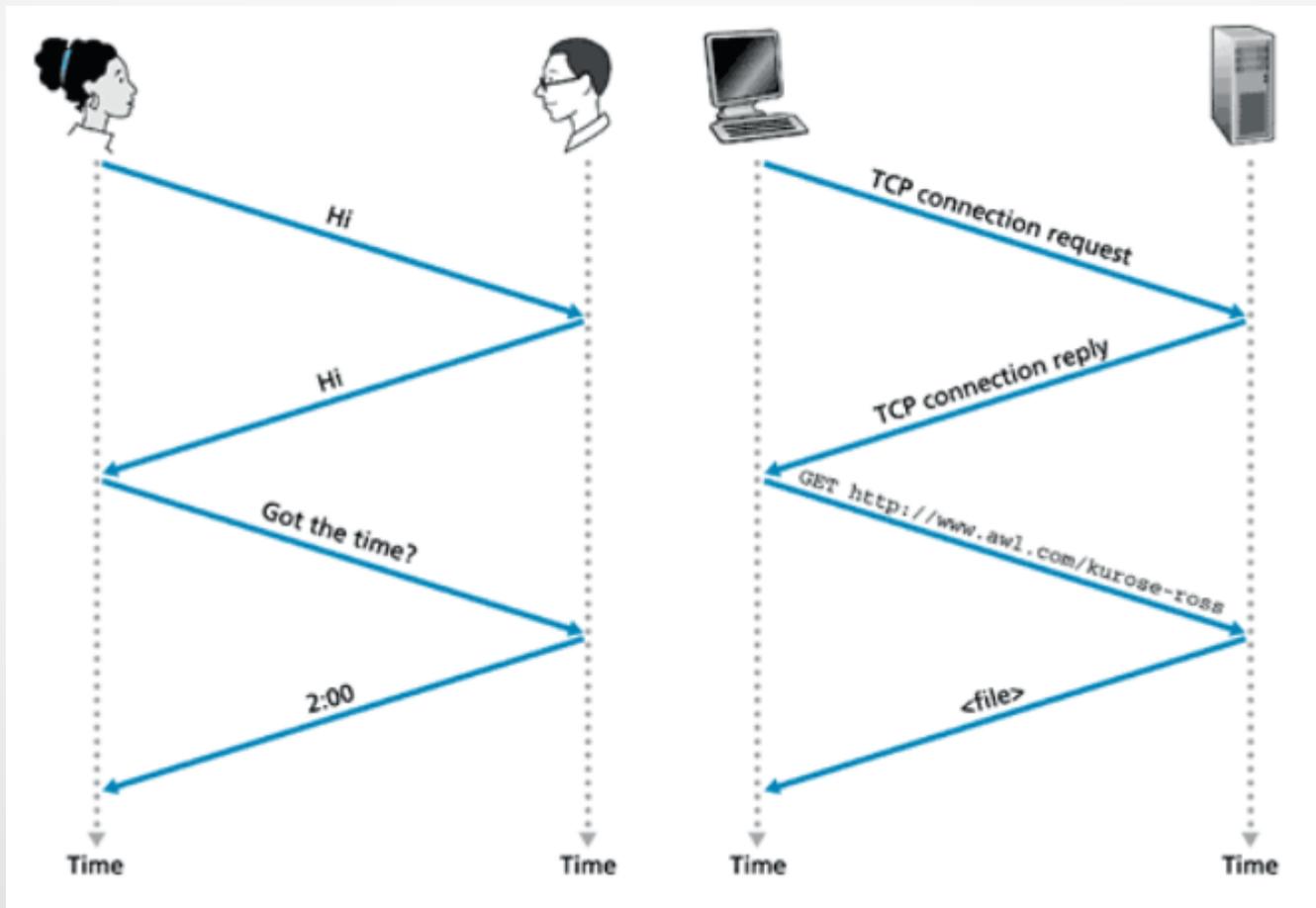
Internet oggi usato nella vendita di beni e servizi. Trasmissione audio video, collaborazione in rete, lavoro, giochi in rete, interazioni sociali.



INTERNET

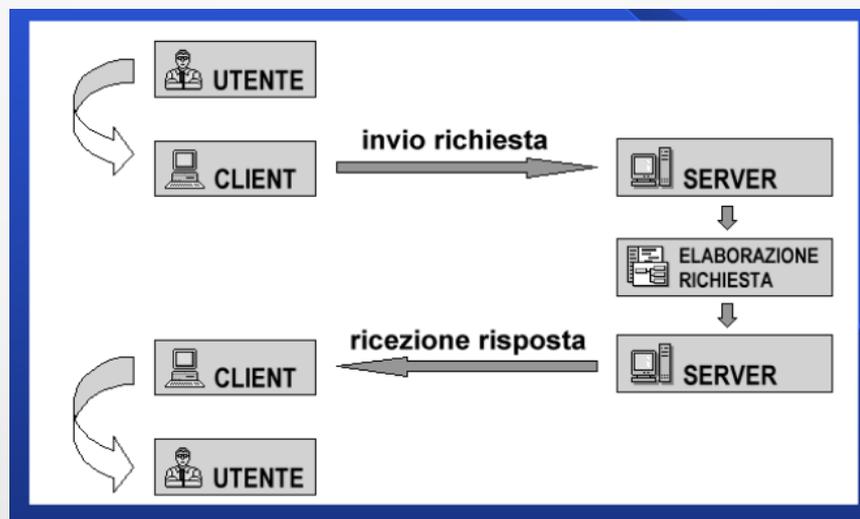
Protocollo di comunicazione

- Insieme di regole (formalmente descritte) che definiscono le modalita' di comunicazione tra due o piu' entita'.



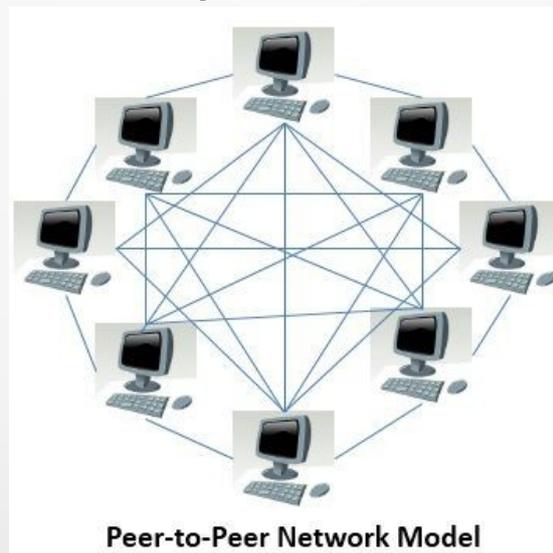
Modello di rete: Client-Server

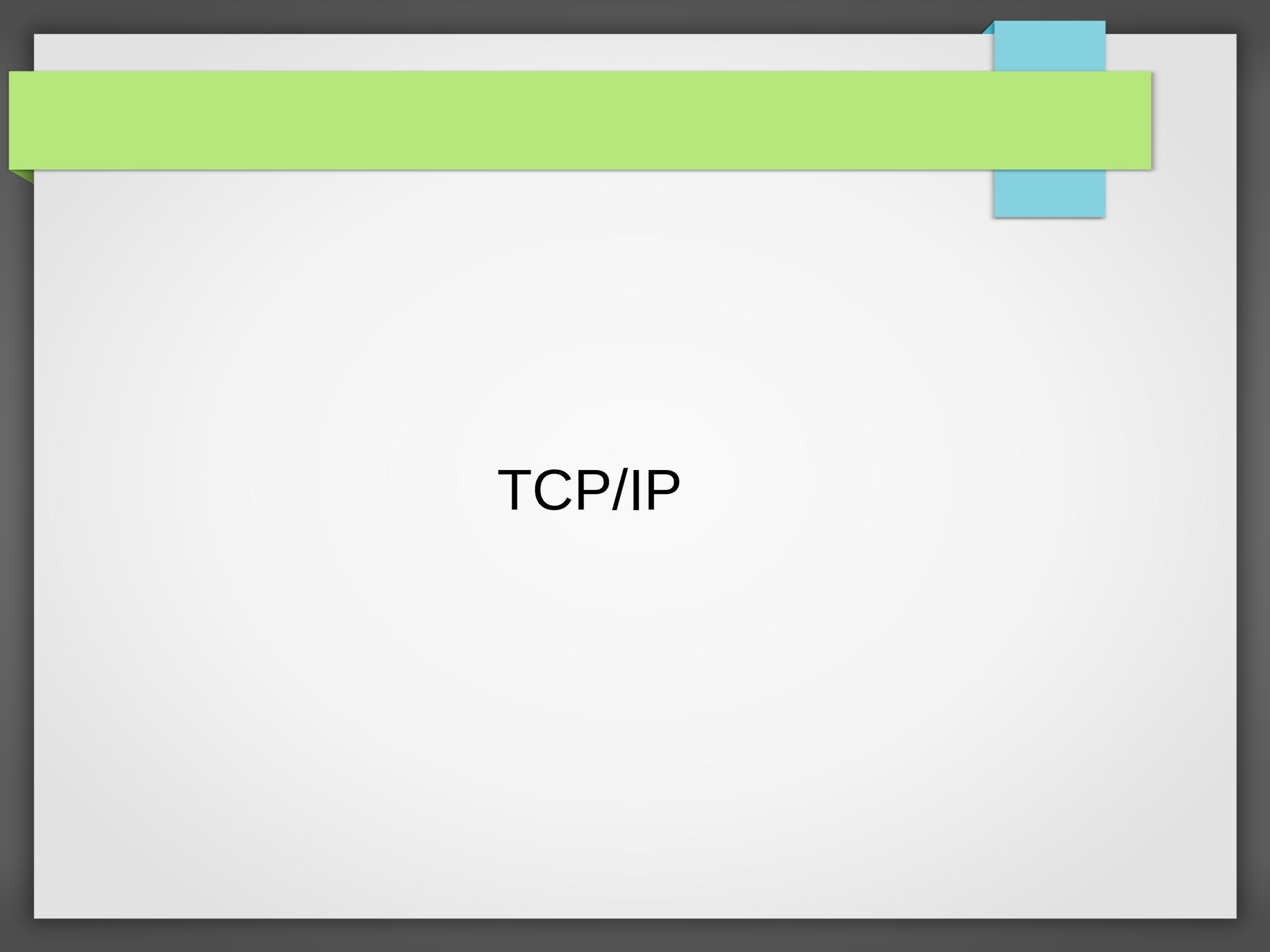
- La maggior parte dei servizi telematici offerti da internet si basano sulla **modalita' di interazione client/server (diverso rispetto al P2P)**.
- Il client e' dotato di un particolare software client in grado di inviare richieste di servizio ad un articolare server. Il client formatta le richieste in modo adeguato e comprensibile al server, usando quindi uno specifico protocollo (es: browser server web)



Modello di rete: P2P Peer-to-Peer

- In questo caso **non c'è una distinzione fra client e server**, **ogni nodo può invece agire sia da client che da server** dipendentemente se il singolo nodo stia richiedendo oppure fornendo dati
- Per diventare parte della rete **P2P dopo il join il nodo deve iniziare a fornire servizi e potrà chiedere a sua volta servizi ad altri nodi** (es: BitTorrent)





TCP/IP

TCP/IP: funzionamento base

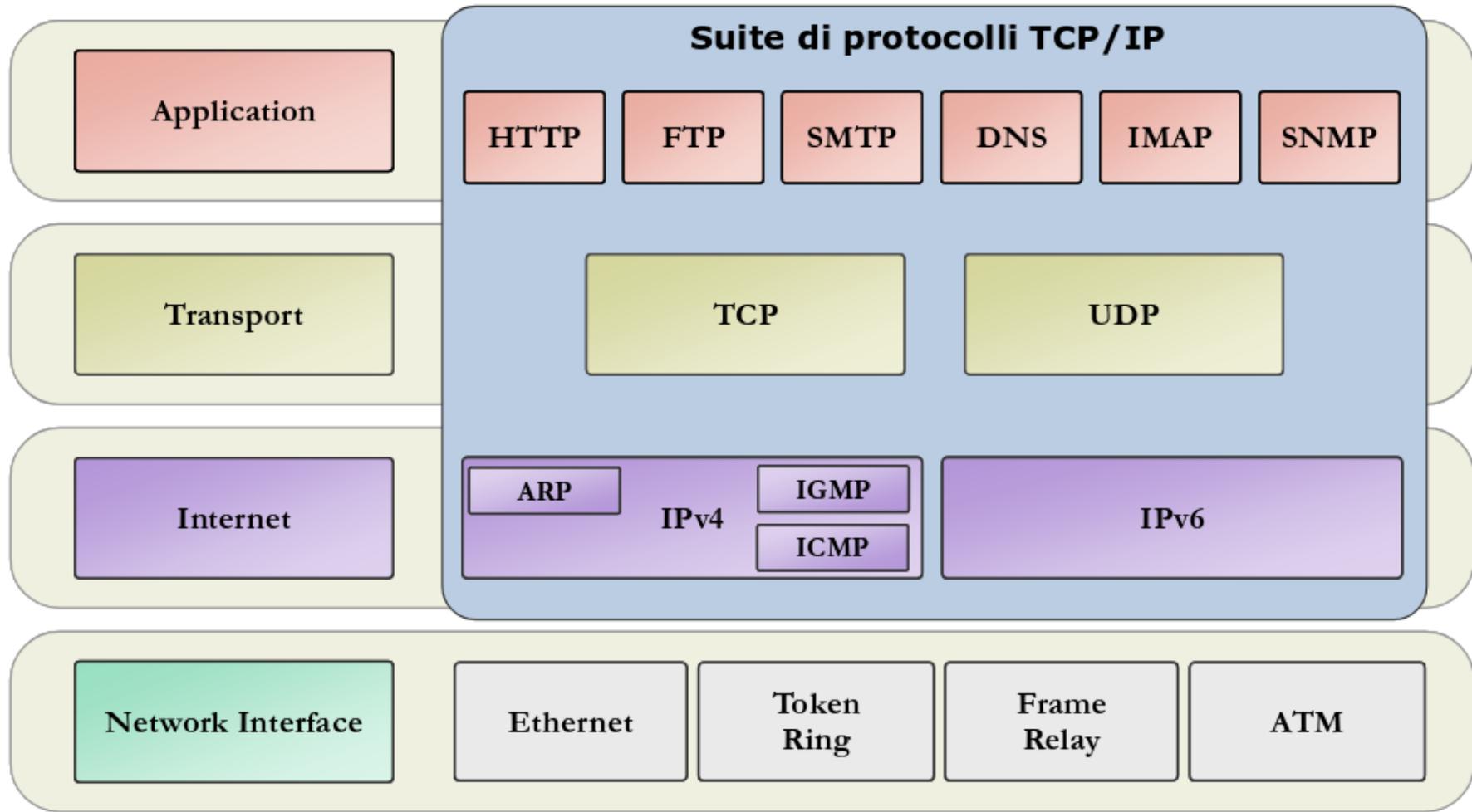
- Un protocollo e' un insieme di regole che chi spedisce dati (mittente) e chi li riceve (destinatario) devo seguire
- Esempio banale di un "protocollo" (non formale) che due persone seguono quando si incontrano ed una chiede l'ora:
 - Bob saluta Alice
 - Alice saluta Bob
 - Bob chiede l'ora
 - Alice dice l'ora
 - Bob ringrazia e saluta
 - Alice saluta

TCP/IP: funzionamento base



Ed alla fine una ricevuta di ricezione (ACK)

Architettura TCP/IP



TCP/IP: funzionamento base

- Idea del **layering**. Ad esempio il protocollo **HTTP e' costruito sopra il TCP**, Un browser eb non si deve preoccupare di come e' implementato il TCP, ma deve solo sapere che funziona.
- **I dati che il livello di trasporto riceve da quello applicativo sono frammentati in pacchetti**. I pacchetti di dati sono ricomposti a destinazione (**ogni pacchetto puo' seguire strade diverse**)
- **Il TCP aggiunge ad ogni pacchetto delle informazioni aggiuntive** come in particolare il numero d'ordine della sequenza di cui il pacchetto fa parte.
- Il pacchetto viene poi passato al livello di rete dove **IP instrada i pacchetti verso l'host di destinazione** nel modo piu' opportuno

Lo STACK TCP/IP: Livello applicativo

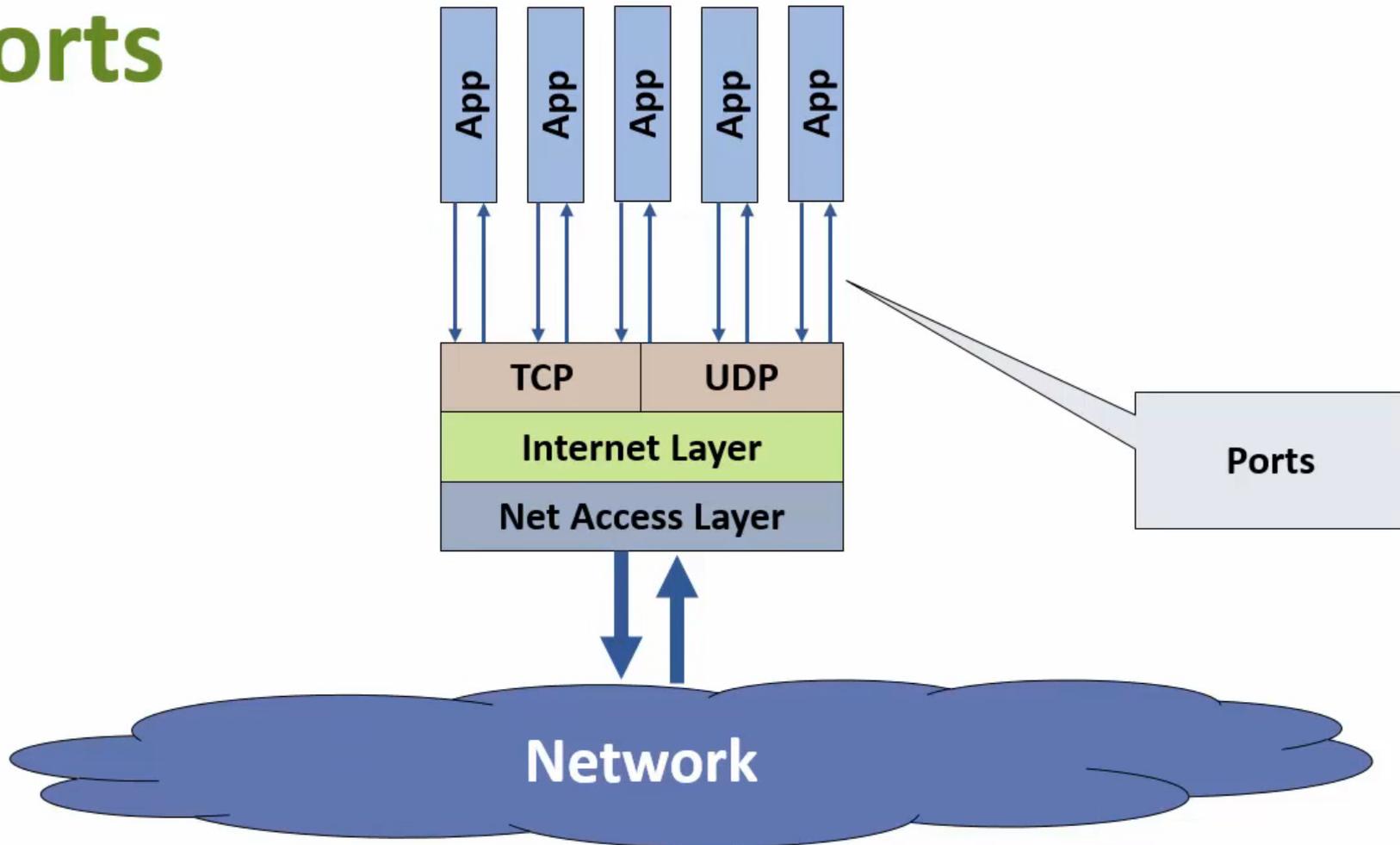
- **Livello applicativo:** troviamo numerosi protocolli come HTTP, FTP, DNS, SMTP, etc etc. A questo livello due applicazioni si scambiano **messaggi** senza preoccuparsi di come questi verranno consegnati.
- Questa e' l'interfaccia con l'utente , ad esempio se consultiamo una pagina web il protocollo gestisce la sessione di interazione fra il nostro browser (client) ed il server web

Lo STACK TCP/IP: Livello di trasporto

- **Livello di Trasporto:** a questo livello troviamo i due protocolli base **TCP** ed **UDP** , a questo livello due hosts si scambiano i segment .
- I protocolli a questo livello **offrono il servizio di trasporto** al livello applicativo, Per gestire ad esempio **piu' sessioni attive contemporaneamente il TCP e l'UDP usano diversi numeri di porta** (porte logiche)
- **TCP**
 - Ad ogni finestra di pacchetti spediti il TCP fa partire un contatore di tempo
 - Chi riceve invia un **ACK** se ha ricevuto il pacchetto
 - Se chi trasmette non riceve un **ACK** prima che scada il tempo (oppure...) . Il trasmettitore si occupa ad esempio di rinviare i dati

Lo STACK TCP/IP: Livello di trasporto

Ports



Lo STACK TCP/IP: Livello di rete

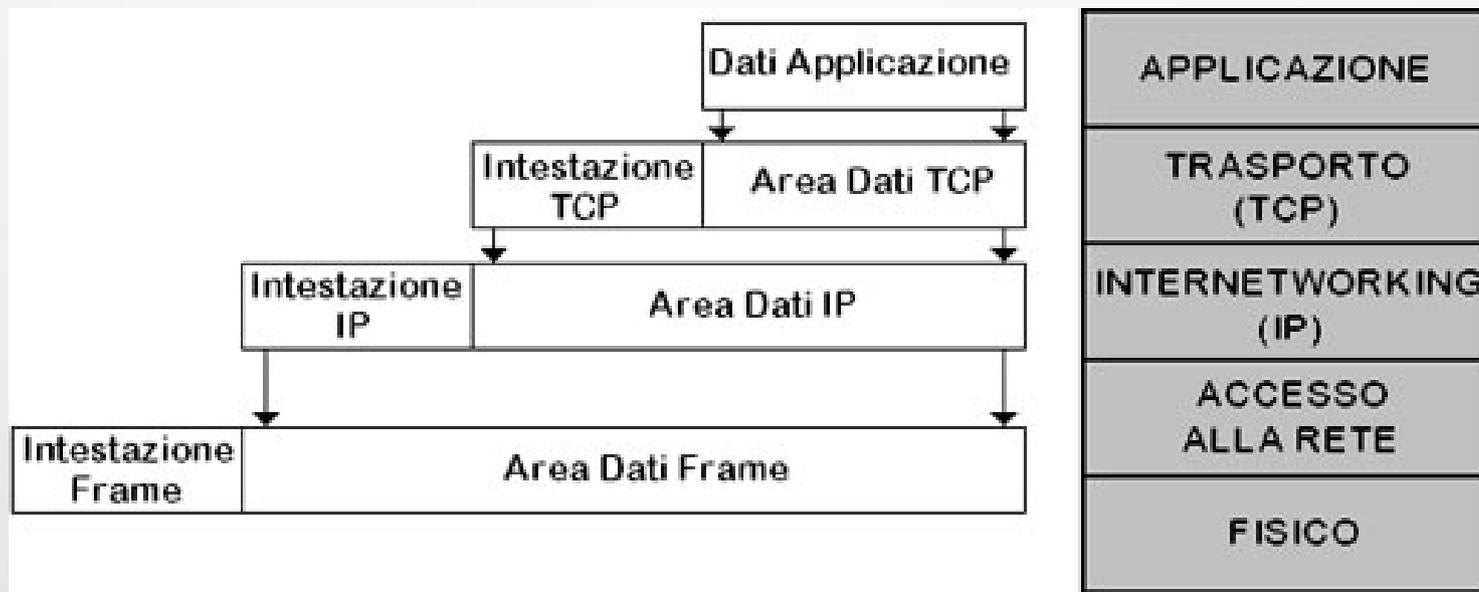
- **Livello di Rete:** a questo livello troviamo il protocollo **IP**. Questo protocollo si occupa dell'indirizzamento e dell'instradamento dei pacchetti fra mittente e destinatario.
- **Indirizzamento:** Ogni nodo e' identificato in modo non ambiguo da un **indirizzo IP**
- **Instradamento:** questa funzionalita' consente di selezionare il percorso migliore da seguire per far transitare i dati dal mittente verso il destinatario

Lo STACK TCP/IP: Livello di accesso alla rete

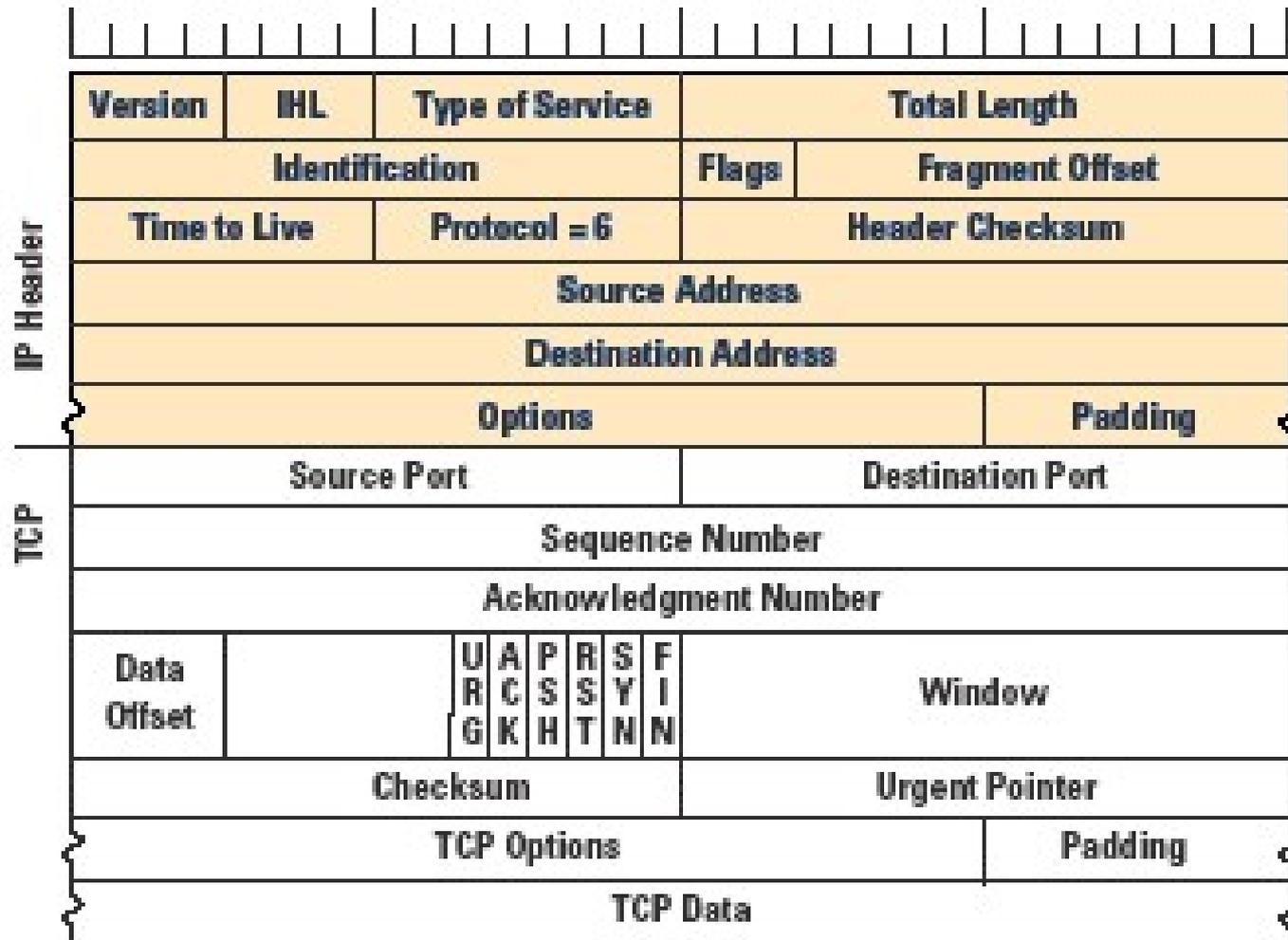
- Il modello TCP/IP, specifica solo che sotto al livello IP ci deve essere un livello di accesso alla rete che si occupi fattivamente di spedire i pacchetti.
- Al livello di collegamento i protocolli decidono come il messaggio debba essere trasferito per ogni tratto del percorso. Quindi ad esempio come andare dal primo host al primo router e così via (**indirizzi MAC, ed ethernet**)
- A livello fisico poi i dati sono convertiti in **segnali elettrici od elettromagnetici o ...**

Lo STACK TCP/IP

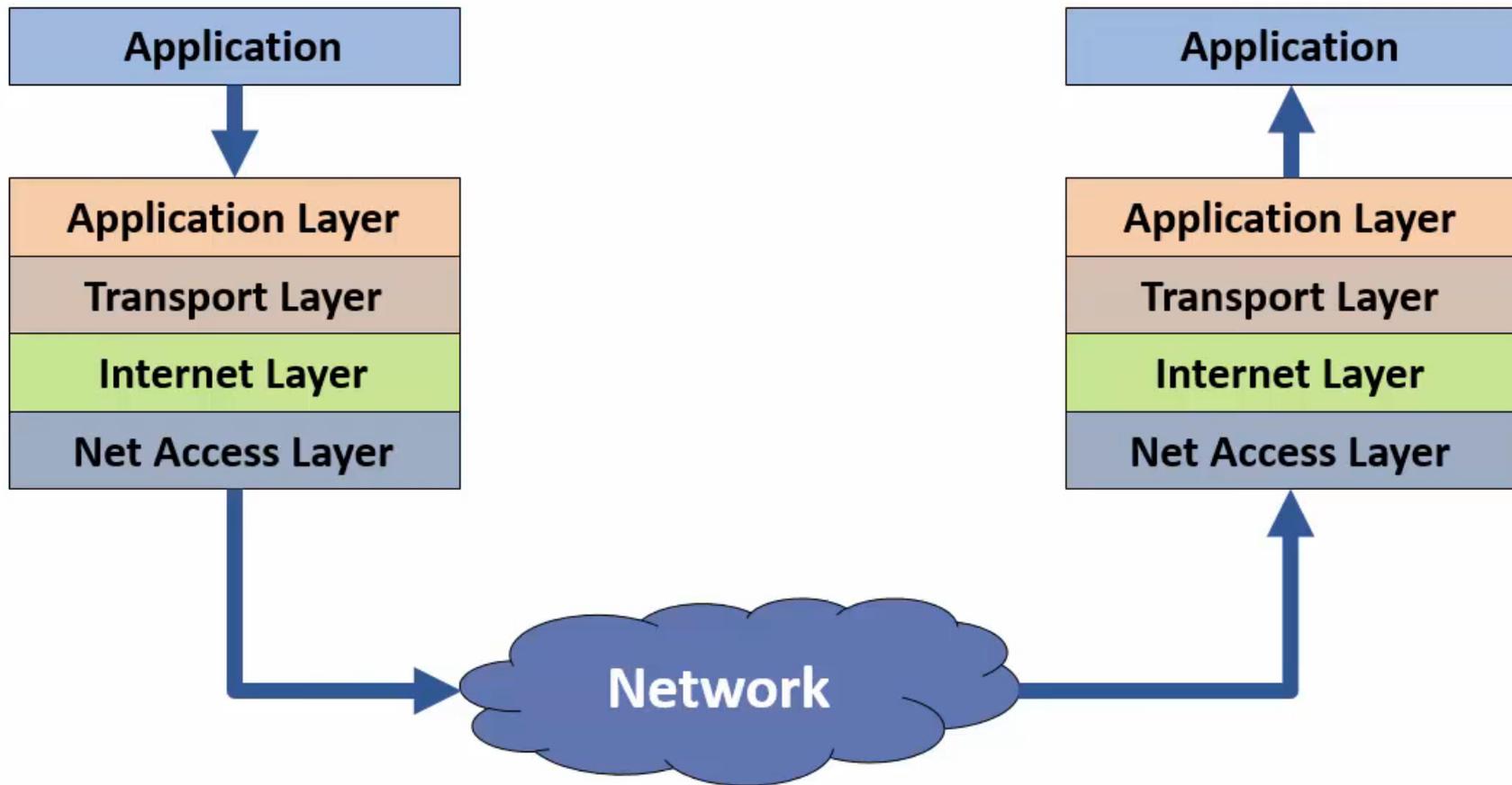
- Quando un'applicazione deve inviare dati, questi vengono passati al livello inferiore, di volta in volta fino a raggiungere la rete fisica sottostante. Durante questo percorso, ogni strato aggiunge informazioni ai dati, fino a creare un "frame di rete" (incapsulamento):

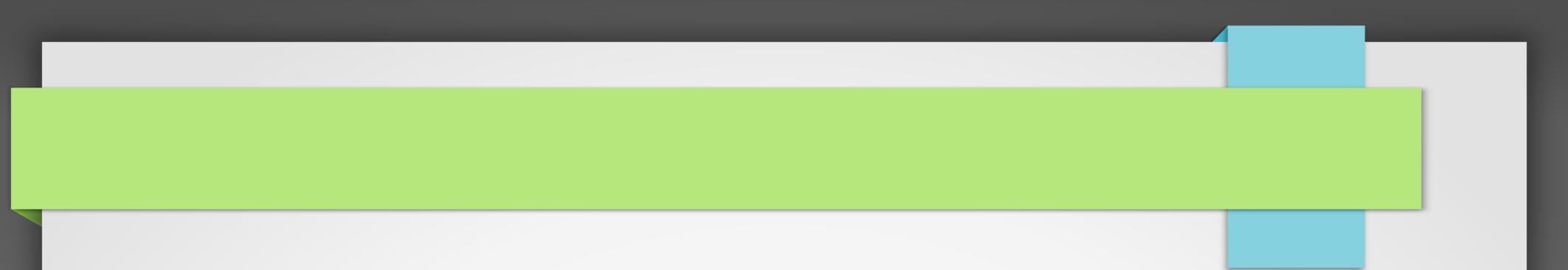


TCP/IP: headers



TCP/IP: headers





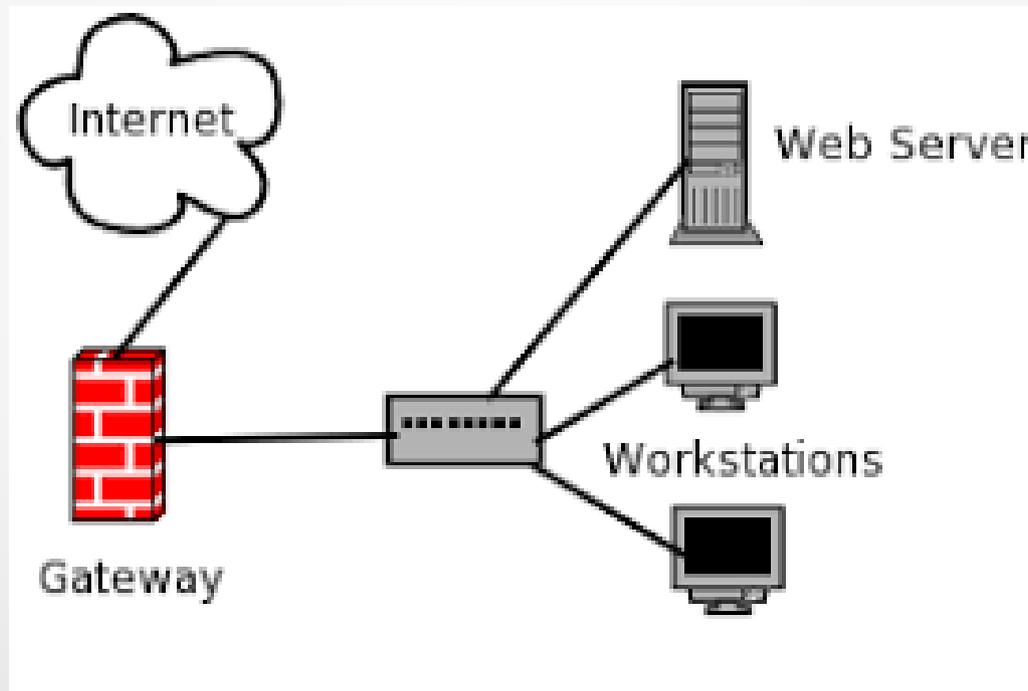
INDIRIZZI IP, DNS, DHCP AND ...

Indirizzi IP

- **Ogni computer collegato ad internet e' identificato dal suo indirizzo IP**, composto da **4 gruppi** di un byte ciascuno (**complessivamente 32-bit**). Ogni numero puo' assumere valori da **0 a 255**
- Ad esempio: **192.167.12.66** (IP statici o Dinamici, IP privati ...)
- L'ultimo numero identifica solitamente un Host, i numeri precedenti la sottorete a cui questo Host appartiene.
- **Il massimo numero di indirizzi IPv4 e' dunque $255*255*255*255$**
- **IPv6: 128-bit** e quindi 2^{128} circa 3.4×10^{38} indirizzi (**IoT: Internet of things**)

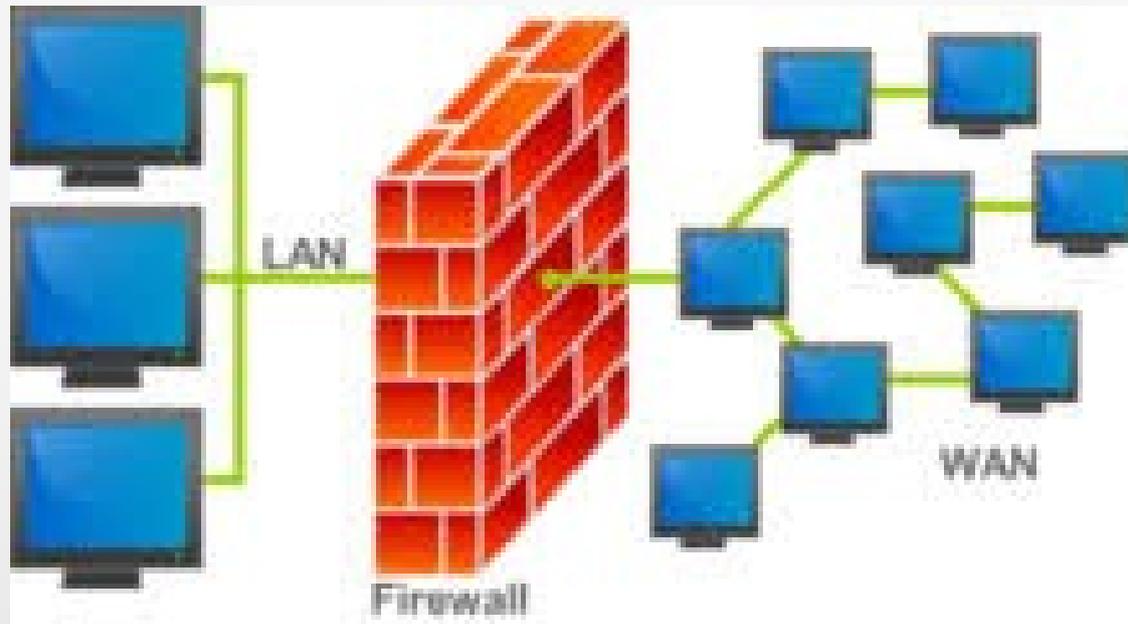
Indirizzi IP: gateway

- Il **gateway** di default viene utilizzato per instradare pacchetti verso altre destinazioni
- Quasi sempre il protocollo **DHCP** viene usato per fornire automaticamente al client l'indirizzo IP del gateway predefinito



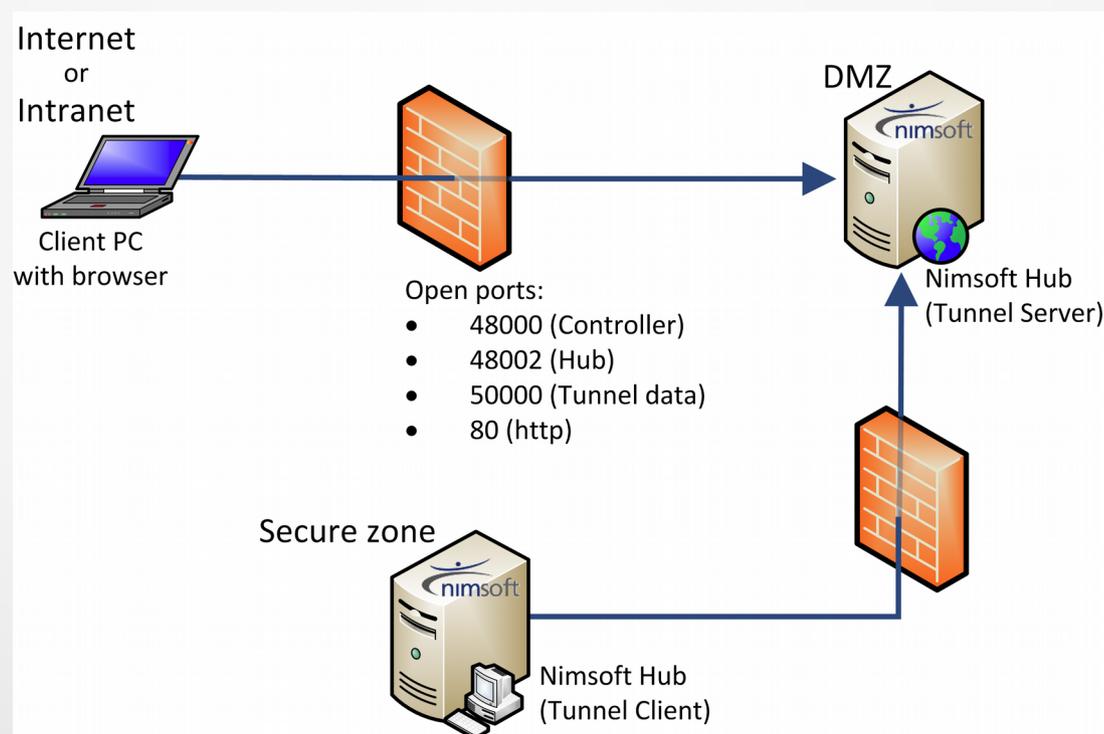
Indirizzi IP: firewall

- Il **firewall** e' un sistema di sicurezza della rete che controlla tutto il traffico in ingresso ed in uscita secondo regole ben definite



Indirizzi IP: DMZ

- Una **DMZ** (demilitarized zone) è una sottorete fisica o logica che contiene ed espone i servizi di un'organizzazione esterna verso una rete non protetta, solitamente una rete più grande come Internet



DNS

- E' difficile per un essere umano memorizzare numeri, molto piu' facile nomi. Ci sono quindi servizi di DNS (**Domain Name System**). Quindi sistemi utili a tradurre in un verso e nell'altro nomi e indirizzi.

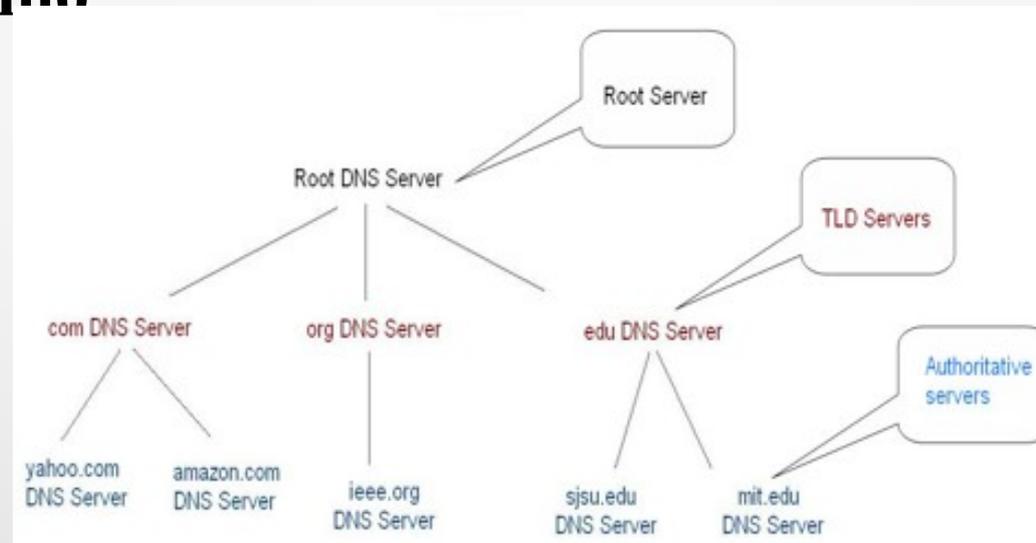
```
redo@eeegw:~$ host www.storchi.org
www.storchi.org has address 82.221.102.244
redo@eeegw:~$ host gw-thch.unich.it
gw-thch.unich.it has address 192.167.12.66
redo@eeegw:~$ host 192.167.12.66
66.12.167.192.in-addr.arpa domain name pointer gw-thch.unich.it.
redo@eeegw:~$
```

DNS

- Ogni host e' dunque identificato dall'utente da un nome simbolico:
 - gw-thch.unich.it
- I nomi sono assegnati univocamente e gestiti amministrativamente in modo gerarchico
- I nomi identificano in modo univoco un host all'interno di un dominio:
 - it e' il dominio
 - unich e' il sotto-dominio all'interno ad it
- I domini principali sono:
 - .gov .edu .com essenzialmente in USA associati al tipo di organizzazione
 - Le varie nazioni invece hanno domini del tipo: .it, .uk, .fr , .de

DNS

- Prima dell'introduzione del sistema DNS la corrispondenza fra indirizzi IP e nomi era gestita dallo SRI-NIC che sostanzialmente manteneva una lista in un file hosts.txt
- In pratica il **DNS e' un database distribuito**. Le informazioni sono infatti distribuite su molti computer, server DNS, **ognuno dei quali e' responsabile di una certa porzione del nome, detta dominio**

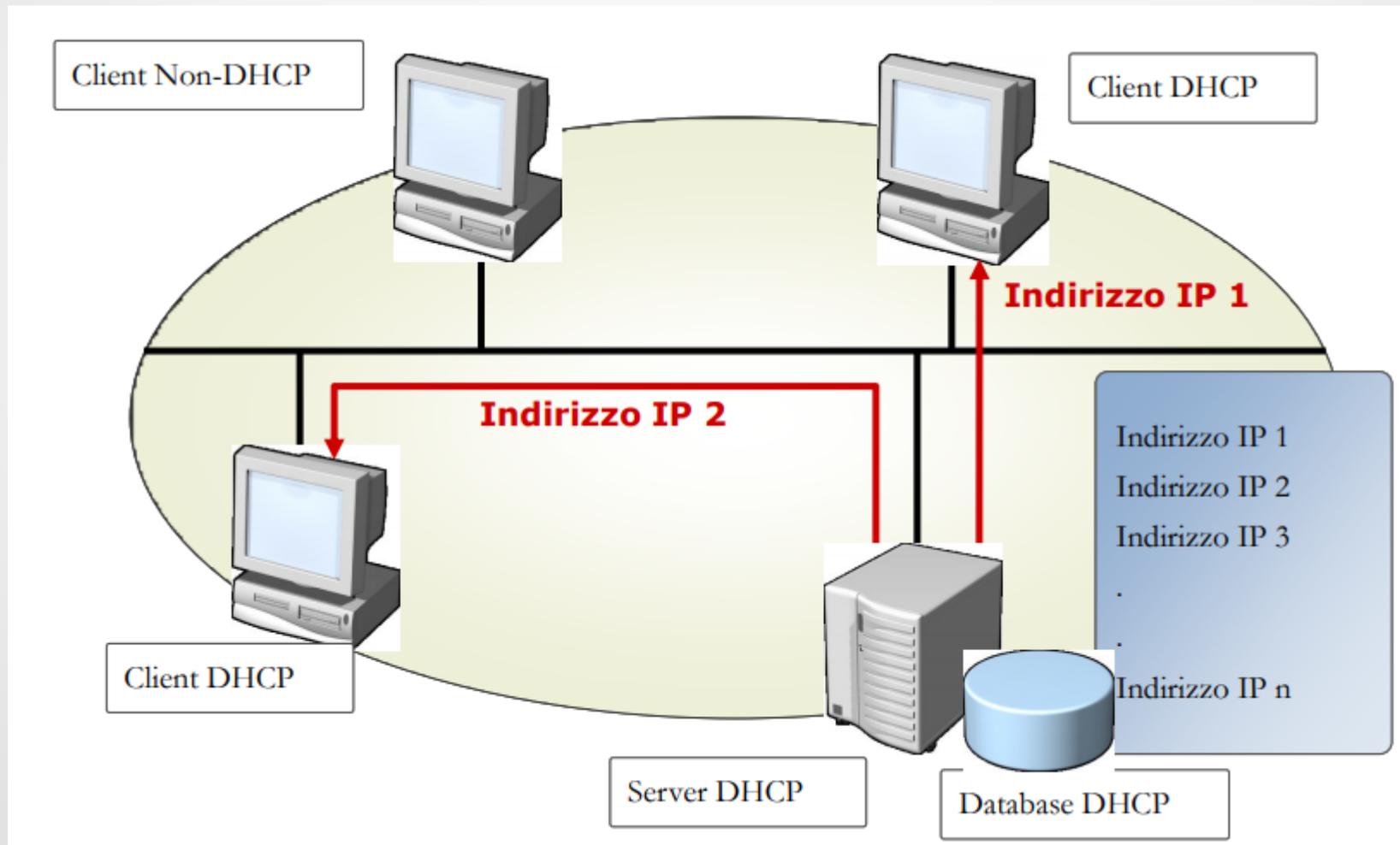


DNS

- I server sono organizzati con una **struttura gerarchica ad albero**
- Al momento della richiesta di un dato indirizzo, ad esempio `www.storchi.org`, il server DNS della “propria rete” controlla se l’indirizzo corrispondente al **nome e’ presente nella cache**.
- Se non e’ presente **contatta i Root servers (quelli che gestiscono le estensioni .org, .it, .edu ...)**, in questo caso `.org` che restituirea’ una lista di **server che gestiscono il dominio storchi.org**
- **Quest’ultimo server restituira’ l’indirizzo IP corrispondente a “WWW” (Domain Name System (DNS) names are "case insensitive")**

DHCP

- Ad esempio il router ADSL che avete a casa



Protocolli ad alto livello

- Vengono usati diversi tipi di protocollo ognuno per ogni specifico servizio:
 - **HTTP (HyperText Transfer Protocol)** Accesso alle pagine ipertestuali (WEB) nell'ambito del WWW (https crittato)
 - **FTP (File Transfer Protocol)** trasferire e copiare file
 - **SMTP (Simple Mail Transfer Protocol)** Spedizione di messaggi di posta elettronica (e-mail) **POP3** per scaricare i messaggi e-mail nel proprio computer . **IMAP utile quando si consultano i messaggi da piu' dispositivi**

Una risorsa in rete e' quindi "identificata" dall'URL:

`http://nomehost.it/index.html`

SMTP esempio

SMTP Protocol Exchange

```
S: 250 SMTPUTF8
C: EHLO example.com
S: 250-mx.google.com at your service, [999.999.999.999]
S: 250-SIZE 35882577
S: 250-8BITMIME
S: 250-AUTH LOGIN PLAIN XOAUTH XOAUTH2 PLAIN-CLIENTTOKEN
S: 250-ENHANCEDSTATUSCODES
S: 250-PIPELINING
S: 250-CHUNKING
S: 250 SMTPUTF8
C: AUTH XOAUTH2 dXN1cj1hbWFnYWtpLnRv...
S: 235 2.7.0 Accepted
```

Specify **Initial Client Response** which is created from username and **access token**

