



ITIS G. Marconi - Catania

Le Reti Informatiche

modulo 3



Prof. Salvatore Rosta

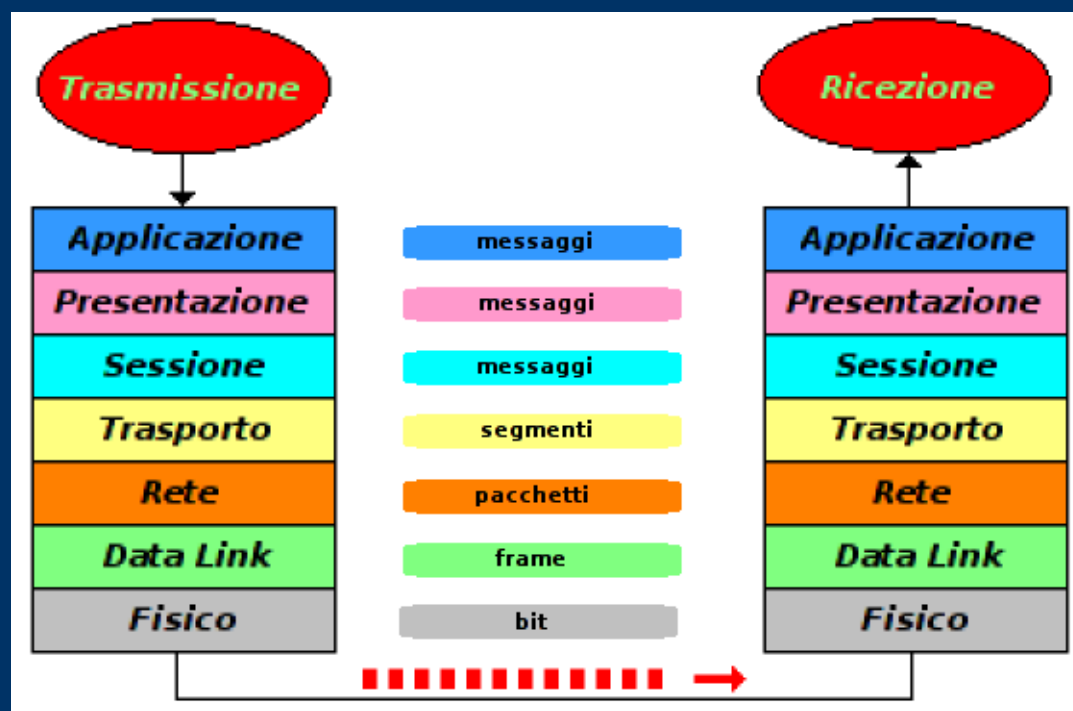
www.byteman.it

s.rosta@byteman.it

Livelli ISO/OSI: 1

Quando 2 computer comunicano i corrispondenti livelli si scambiano **virtualmente** dei dati che hanno dei formati particolari.

La comunicazione **effettiva** avviene sempre a livello fisico.



Livelli ISO/OSI: 2

Le conversazioni effettuate tra livelli corrispondenti vengono anche chiamate **processi peer**, mentre le entità logiche coinvolte a livello **n** sono dette **peer entity**.

Questi processi avvengono secondo determinate regole chiamate **protocolli**.

Quindi in corrispondenza di ogni Livello si avranno i protocolli di:
Applicazione, Presentazione, Sessione, Trasporto, Rete, Data-link, Fisico.

Livelli ISO/OSI: 3

Il **modello** standard ISO/OSI non definisce i protocolli e pertanto non può essere considerato come **architettura di rete**.

Nessuno utilizza in pratica tutti i livelli ISO/OSI, ma lo **stack** di comunicazione dei telefoni cellulari GSM è *quasi* uno stack ISO/OSI.

E' però un modello che è importante conoscere quando si intendono studiare reti di telecomunicazione diverse da Internet.

Serve da riferimento per gli sviluppatori di protocolli e per comprendere ed interpretare i protocolli odierni.

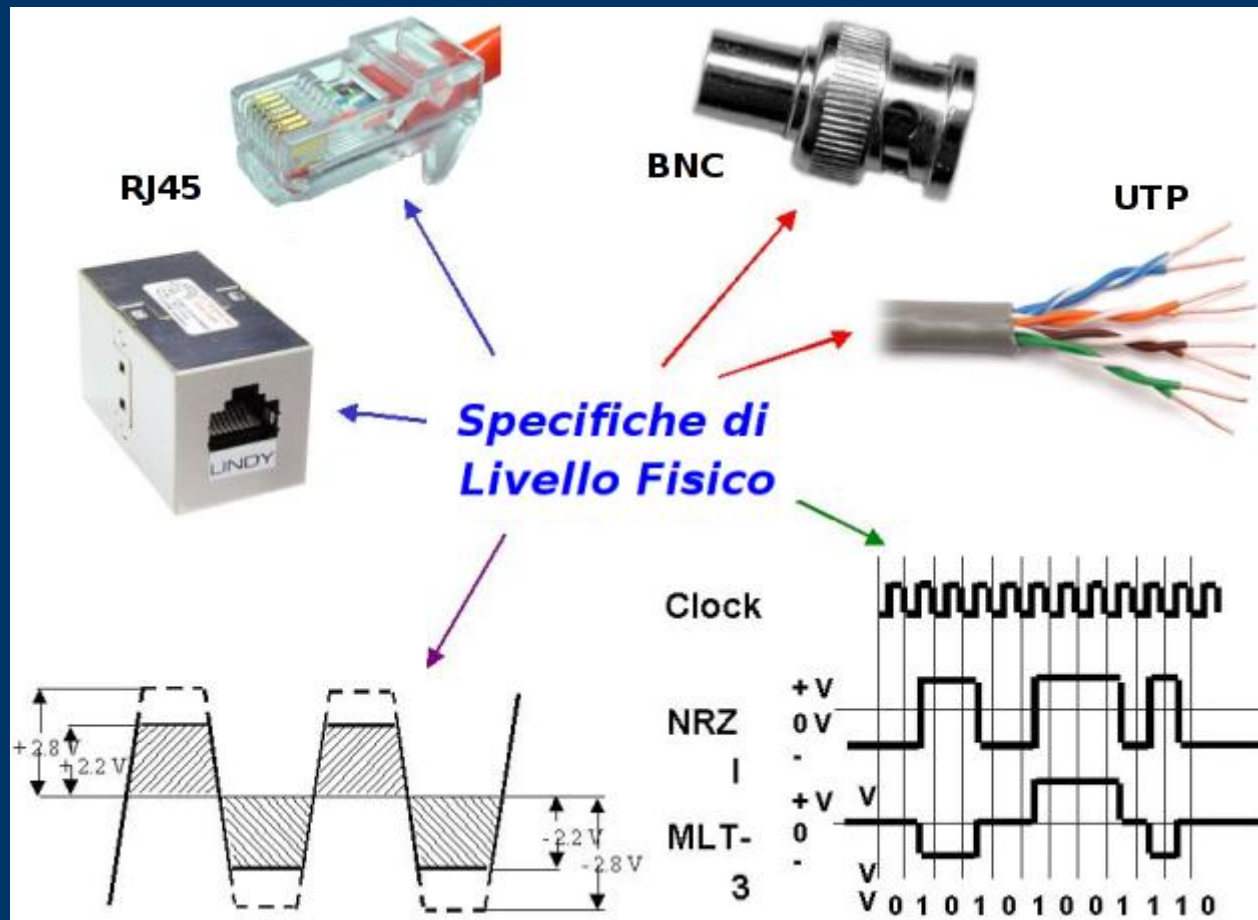
Il livello Fisico: 1

Riguarda la trasmissione/ricezione dei **BIT** sul canale fisico di trasmissione (**fili**, **onde radio**, **luce**, **laser**).

Coinvolge aspetti di tipo:

- elettrico (linee comunicazione, propagazione onde)
- comunicazione (modalità: simplex, half-duplex, full-duplex; segnali di controllo, temporizzazione,...)
- meccanico (standard dei connettori, ...)
- tecnologico (Ethernet, Arcnet, token ring, ...)

Il livello Fisico: 2



Il livello Fisico: 3

Simplex: trasferimento dati unidirezionale.

Come nelle tradizionali trasmissioni radio, è possibile un trasferimento dati in una sola direzione: dalla RAI all'ascoltatore.

HalfDuplex: trasferimento dati bidirezionale alternato.

Come i classici walkie talkie, ricetrasmittitori con i quali quando si parla non si ascolta e viceversa.

FullDuplex: trasferimento dati bidirezionale contemporaneo.

Come avviene con il telefono: è possibile parlare ed ascoltare simultaneamente ...

Il livello Data-link: 1

Riguarda la trasmissione/ricezione di raggruppamenti di BIT chiamati **FRAME**.

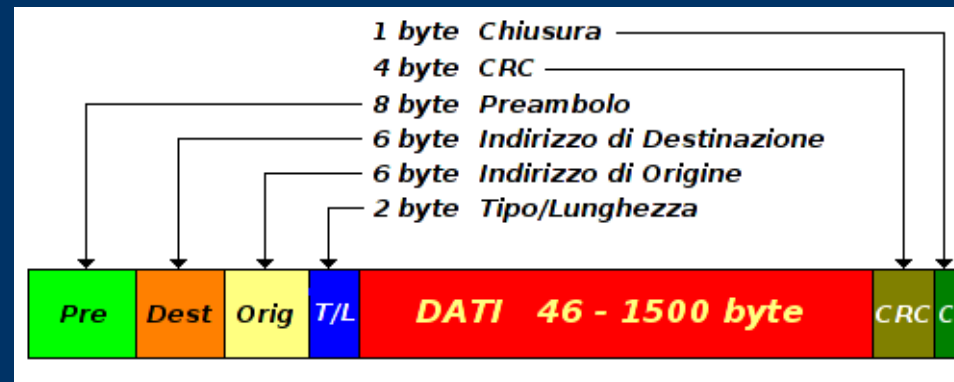
Questo livello è chiamato a svolgere diverse funzioni:

- una buona interfaccia al livello di rete
- il raggruppamento dei bit del livello fisico in frame
- la gestione degli errori di trasmissione
- la regolazione del flusso in modo che i riceventi lenti non siano travolti dai frame dei mittenti rapidi.

Lo scopo di questo livello è la **prevenzione degli errori** e la formattazione in frame è uno dei mezzi per raggiungere questo obiettivo.

Il livello Data-link: 2

Struttura di un tipico frame di dati Ethernet



I dati vengono **“imbustati”** e contrassegnati con elementi che ne individuano l'origine e la destinazione.

Il preambolo e la chiusura consentono la sincronizzazione del tutto, mentre il CRC garantisce l'integrità dei dati.

Il livello Rete: 1

Riguarda la trasmissione/ricezione di FRAME reimpastati e denominati **PACCHETTI**.

In questo livello vengono gestiti, sostanzialmente, l'instradamento dei pacchetti ed il loro controllo di sequenza.

Per l'instradamento vengono usate apposite tabelle di **routing** che, a seconda dei requisiti della rete, possono essere **statiche** o **dinamiche**.

Il livello Rete: 2

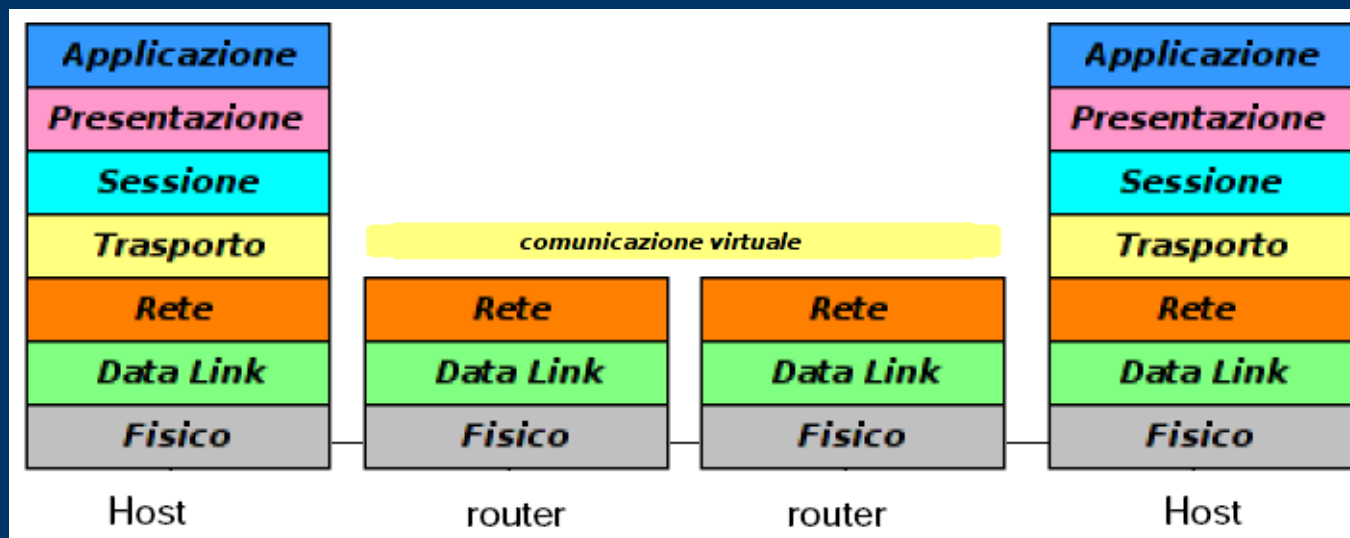
Funzionalità del livello Rete:

- Definisce un sistema uniforme di **indirizzamento**.
- Controlla il cammino ed il flusso di pacchetti (algoritmi di **routing**).
- Gestisce e controlla la **congestione** della rete.
- Gestisce l'**accounting** dei pacchetti sulle reti a pagamento, quindi la contabilità.
- Implementa l'interfaccia necessaria alla comunicazione tra reti di tipo diverso (**internetworking**).
- Gestisce la diffusione di messaggi a più destinazioni (**multicast**).

Il livello Trasporto: 1

Riguarda la trasmissione/ricezione di **SEGMENTI** tra i vari **computer host**, non ci sono più i router di mezzo...

E' il primo dei livelli che consentono una comunicazione virtuale diretta tra due host.



Siamo finalmente fuori dalla **sottorete di comunicazione** costituita dai tre livelli inferiori.

Il livello Trasporto: 2

Il livello di trasporto deve eventualmente sopperire alla mancanza di affidabilità del livello di rete (pacchetti persi, duplicati, invertiti ecc.)

Funzionalità principali:

1. **Servizio orientato alla connessione**: accetta dati dal livello superiore, li spezza in segmenti e li trasmette al destinatario, assicurando per lo meno l'ordine corretto di ricomposizione, e se possibile un servizio privo di errori ed il controllo di flusso.
2. **Servizio senza connessione**: fornisce il recapito dei messaggi senza garanzie.

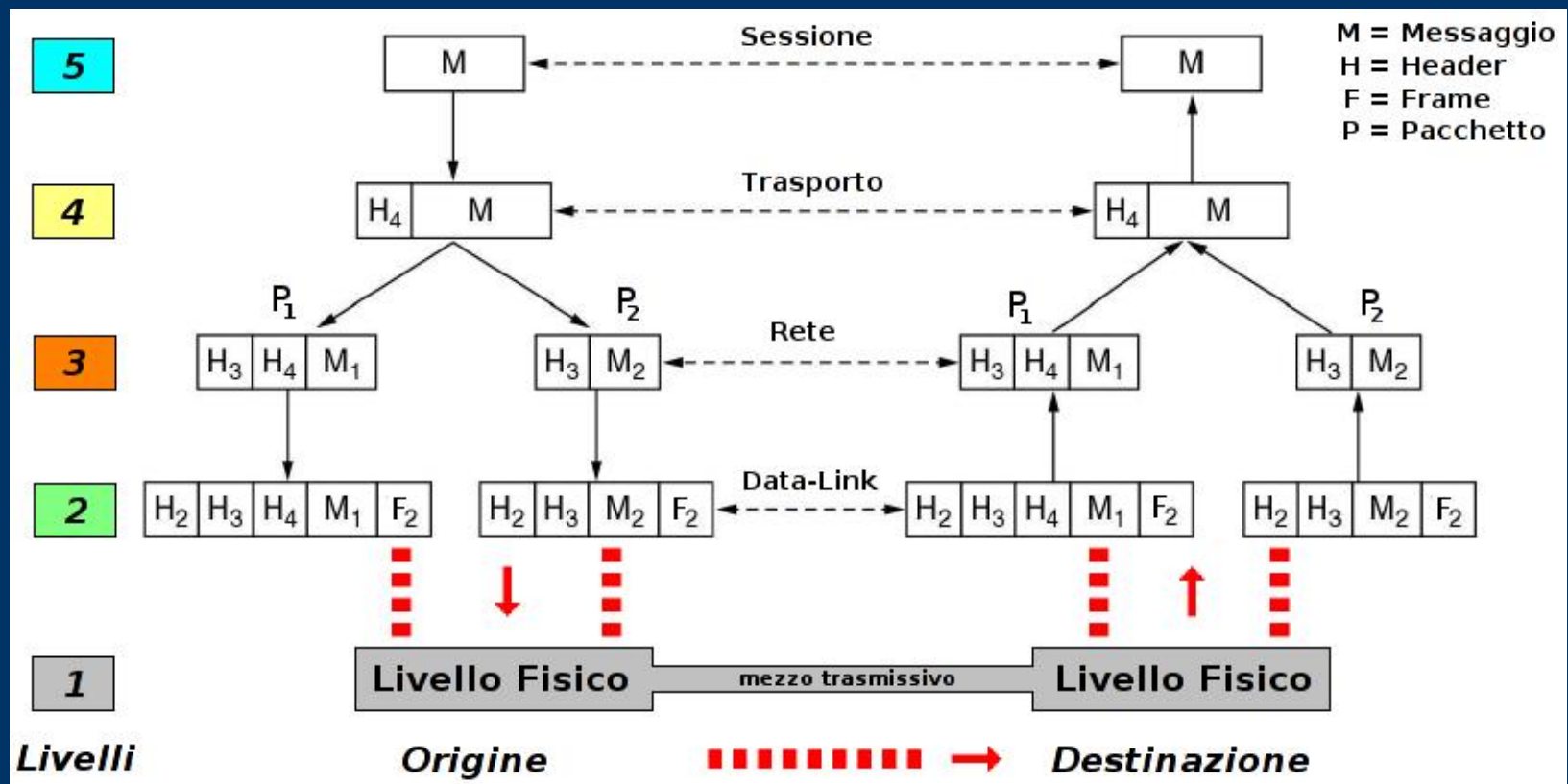
Il livello Sessione: 1

Per ogni **login** (sessione) vengono negoziate e stabilite connessioni tra processi o applicazioni su host diversi.

Le funzionalità di questo livello sono:

- Controllare il dialogo tra due macchine: non sempre la comunicazione è full-duplex; se non lo è questo livello tiene traccia di chi è il turno attuale.
- Gestire il controllo del **token** nel dialogo tra più entità (ruolo del moderatore nei faccia a faccia televisivi).
- Gestire la sincronizzazione nel trasferimento dei dati (es. checkpoint).

Il livello Sessione: 2



Schema funzionale del percorso dal livello di Sessione al livello Fisico e viceversa.

Il livello Presentazione: 1

Traduzione dei dati che viaggiano sulla rete in un formato astratto.

- La diversità dei sistemi operativi può tradursi in **formati di file** incompatibili. Questo livello dovrebbe integrare queste diversità in modo da renderle trasparenti alle applicazioni di rete.
- La **codifica** dei dati ai fini della sicurezza (crittografia-decrittografia).
- La **compressione** dei dati ai fini della riduzione del volume di traffico.
- Nascondere le differenze hardware di dispositivi assimilabili (terminali-stampanti).

Il livello Applicazione: 1

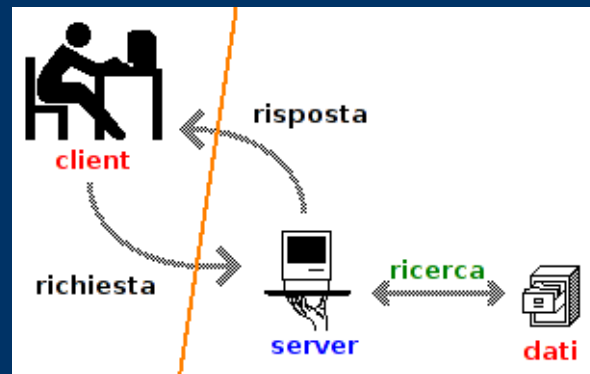
Lo strato applicazione comprende una varietà di protocolli utilizzati dagli utenti o dalle applicazioni.

Alcuni esempi di applicazioni sono:

- DNS (Domain Name Server)
- FTP (File Transfer Protocol)
- Posta elettronica (SMTP, POP, IMAP)
- World Wide Web (protocollo HTTP)
- Applicazioni Multimediali (protocollo RTP)

Il modello Client-Server: 1

Assieme al modello ISO/OSI occorre tenere presente anche il **modello client/server**, che serve per la progettazione degli applicativi di rete.



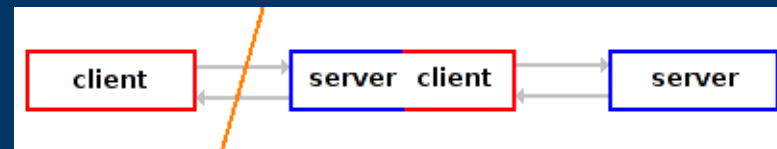
Si basa sulla premessa che le comunicazioni di rete consistono di 2 estremità: **lato client** e **lato server**.

Il lato client richiede informazioni al server, e questo risponde alle richieste del client.

Il modello Client-Server: 2

Un' applicazione di rete completa richiede **2 programmi**, uno con il ruolo di client ed uno con il ruolo di server.

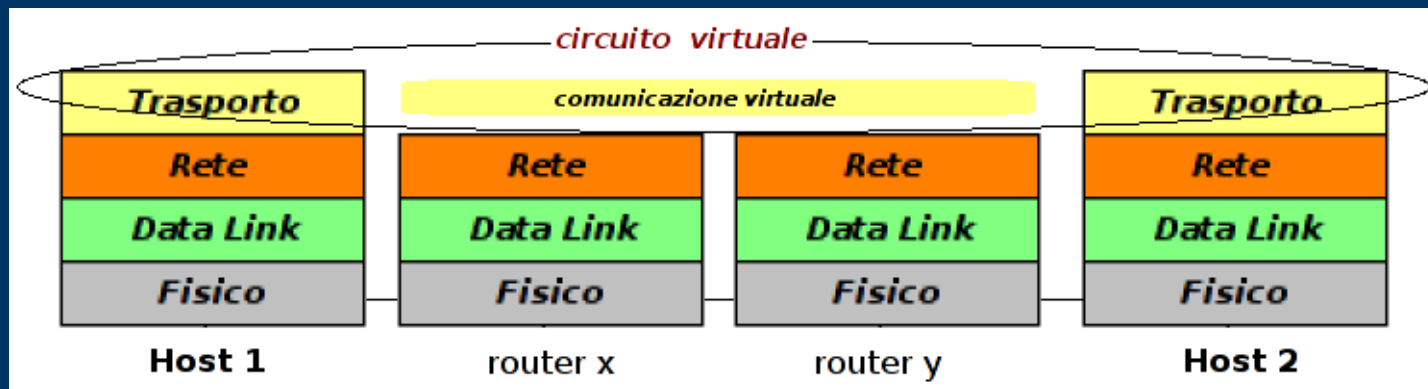
E' però possibile, e talvolta necessario, che un singolo programma svolga entrambe le funzioni, infatti se un server non riesce a soddisfare la richiesta del proprio client dovrà funzionare da client nei confronti di un altro server a cui richiederà le informazioni di cui non dispone.



Il **modello client-server** suddivide le problematiche di progettazione del software in **moduli** ben definiti.

I circuiti virtuali: 1

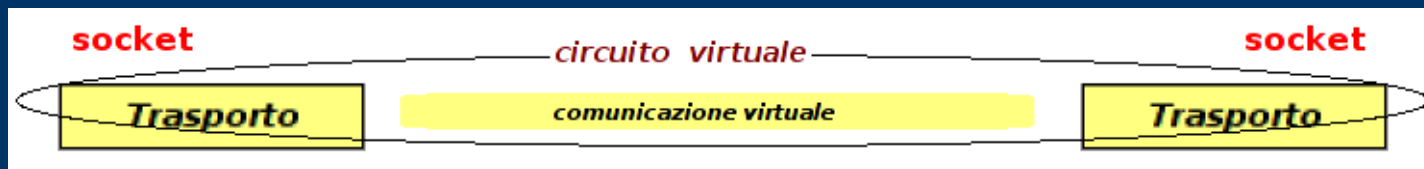
I livelli di rete consentono una conversazione virtuale fra livelli su computer differenti. Una conversazione virtuale fa uso di un circuito (o **connessione**) virtuale.



Il modello client-server lavora sui **circuiti virtuali** come se fossero delle vere connessioni punto a punto (point-to-point).

I circuiti virtuali: 2

Ciascuna estremità di un circuito virtuale viene chiamata **socket**.



Il socket che richiede la connessione è il **lato client** della connessione ed il software che richiede la connessione viene denominato **applicazione client**.

Il socket che riceve la richiesta di connessione è il **lato server** della connessione ed il software che si trova da quella parte viene denominato **applicazione server**.

I circuiti virtuali: 3

Un'applicazione (o processo) **server**, generalmente, si inizializza e poi va in stato di riposo, spendendo la maggior parte del tempo in **attesa di una richiesta** proveniente da un client. Fornisce servizi specifici che sono utili agli utenti di una rete:

- database
- email
- time server
- file server

Un'applicazione **client** trasmette prima una richiesta per una connessione al server e poi richiede dei servizi attraverso la connessione:

- ora del giorno
- file

I circuiti virtuali: 4

Un server è chiamato **iterativo** quando gestisce ogni richiesta individualmente.

Caso tipico è il **time server** che quando riceve una richiesta di ora esatta invia una risposta immediata e non gestisce nessun'altra richiesta finchè non ha soddisfatto quella in corso.

I server iterativi sono semplici ma devono svolgere servizi di breve durata.

Un server è chiamato **concorrente** quando crea un processo separato per gestire ciascuna richiesta di servizio.

Caso tipico è il **file server** che per ogni richiesta di file lancia il processo corrispondente e si rimette in attesa per una nuova richiesta di servizio.

I server concorrenti sono più complessi e richiedono un sistema operativo multitasking e svolgono servizi la cui durata è sconosciuta o non prevedibile a priori.