

Reti di calcolatori: Introduzione e ISO/OSI

Francesco Brun

Sistemi elettronici e informatici in ambito di imaging – II



Obiettivi

- Capire come **due o più calcolatori possono comunicare tra loro**
- La possibilità di far comunicare i calcolatori è alla base di **Internet**
- Grossolanamente, l'obiettivo sarà capire come funziona Internet
- Una rete di calcolatori è comunque indipendente da Internet
- Talvolta si usa il termine **intranet** per indicare la rete interna di un'organizzazione
- I principi di funzionamento di un intranet e di Internet sono gli stessi
- Vedremo solo i concetti principali delle reti di calcolatori...

Definizione di rete

- Una **rete di calcolatori** è un insieme di due o più calcolatori connessi tra loro
- I calcolatori sono chiamati “nodi” o host della rete
- Per essere precisi, “nodo” non coincide con “computer”
- Molti computer hanno in realtà più dispositivi di rete
 - ✓ (es. un laptop ha la scheda di rete cablata e l’antenna per la rete WiFi).
 - ✓ (es. uno smartphone ha l’antenna WiFi e l’antenna 2G/3G)

Reti di calcolatori: perchè?

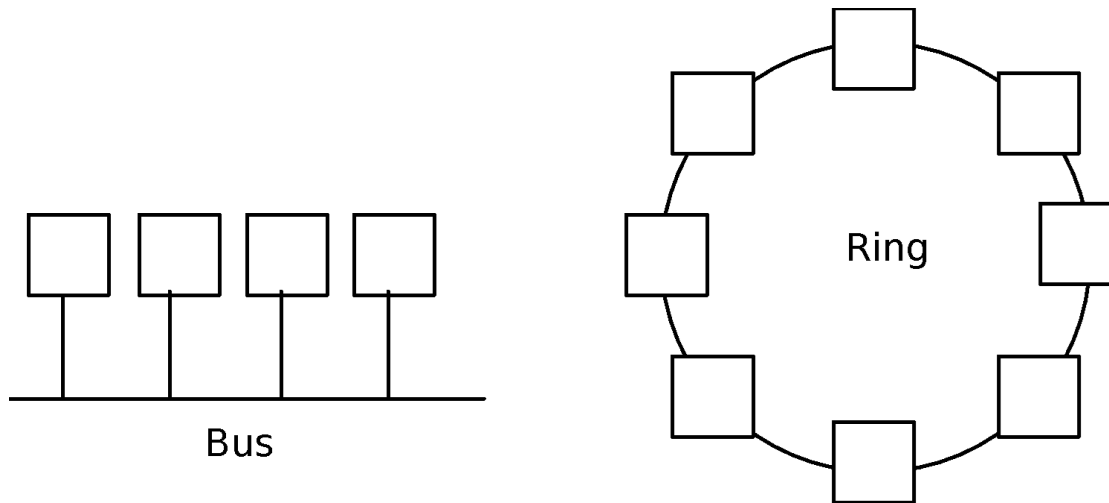
- Vi sono due motivi principali:
- **Condividere risorse**: primo motivo che ha spinto a creare le reti e la prima risorsa che si è voluto condividere è stata la **stampante**.
 - ✓ oggi è comune condividere spazio di archiviazione
 - ✓ si condivide anche l'accesso ad Internet
- **Comunicazione**: i calcolatori connessi possono essere molto distanti tra loro
 - ✓ grazie alle reti è nata la **posta elettronica** (*electronic mail* o email)
 - ✓ ... ma anche il **commercio elettronico** (es. eBay, Amazon, ...)
 - ✓ ... i **social network** (es. Facebook, Instagram, LinkedIn, ...)
 - ✓ ... servizi al cittadino (es. Carta Regionale dei Servizi, ...)

Classificazione delle reti

- Le reti si possono classificare in molti modi
- Dipende da quale aspetto si vuole evidenziare
- Distinzioni interessanti sono quelle basate su:
 - ✓ **tecnologia di trasmissione** (reti *broadcast* e reti *punto a punto*)
 - Essenzialmente ci si chiede se “tutti vedono tutti” o “solo i vicini”
 - ✓ **dimensione** (reti *LAN* e reti *WAN*)
 - Essenzialmente ci si chiede chi è il proprietario dei “cavi di rete”

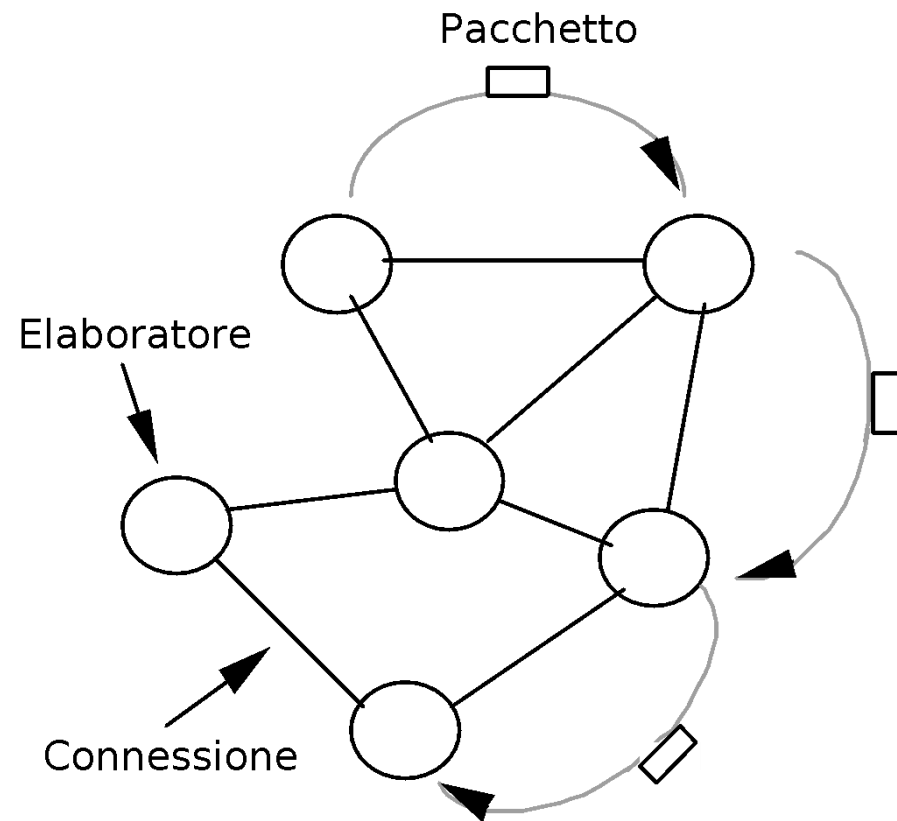
Reti broadcast

- **Broadcast:** vi è un unico "canale" di comunicazione condiviso da tutti
- Il canale può essere ad esempio a **bus** o ad **anello (topologia della rete)**
- Brevi messaggi inviati da un elaboratore sono ricevuti da tutti gli altri
 - ✓ "tutti possono parlare con tutti"
- E' **necessario un indirizzo** per specificare il corretto destinatario



Reti *point-to-point*

- **Punto a punto (*point-to-point*)** in cui le connessioni sono effettuate a coppie
- I nodi della rete “si parlano solo se vicini”
- Da sorgente a destinazione si possono attraversare uno o più nodi intermedi
- Spesso esistono più cammini alternativi
 - ✓ sono necessari algoritmi di **instradamento** (*routing*)
 - ✓ sono necessari dispositivi di instradamento (**router**)



Classificazione per dimensione: reti LAN

- LAN è acronimo di *Local Area Network* (LAN)
- In genere sono reti possedute da un'organizzazione (**reti private**)
- Hanno un'estensione che arriva fino a qualche km
- I cavi di rete si distendono nell'ambito di un singolo edificio o campus
- Di norma una LAN **non posa cavi sul suolo pubblico**

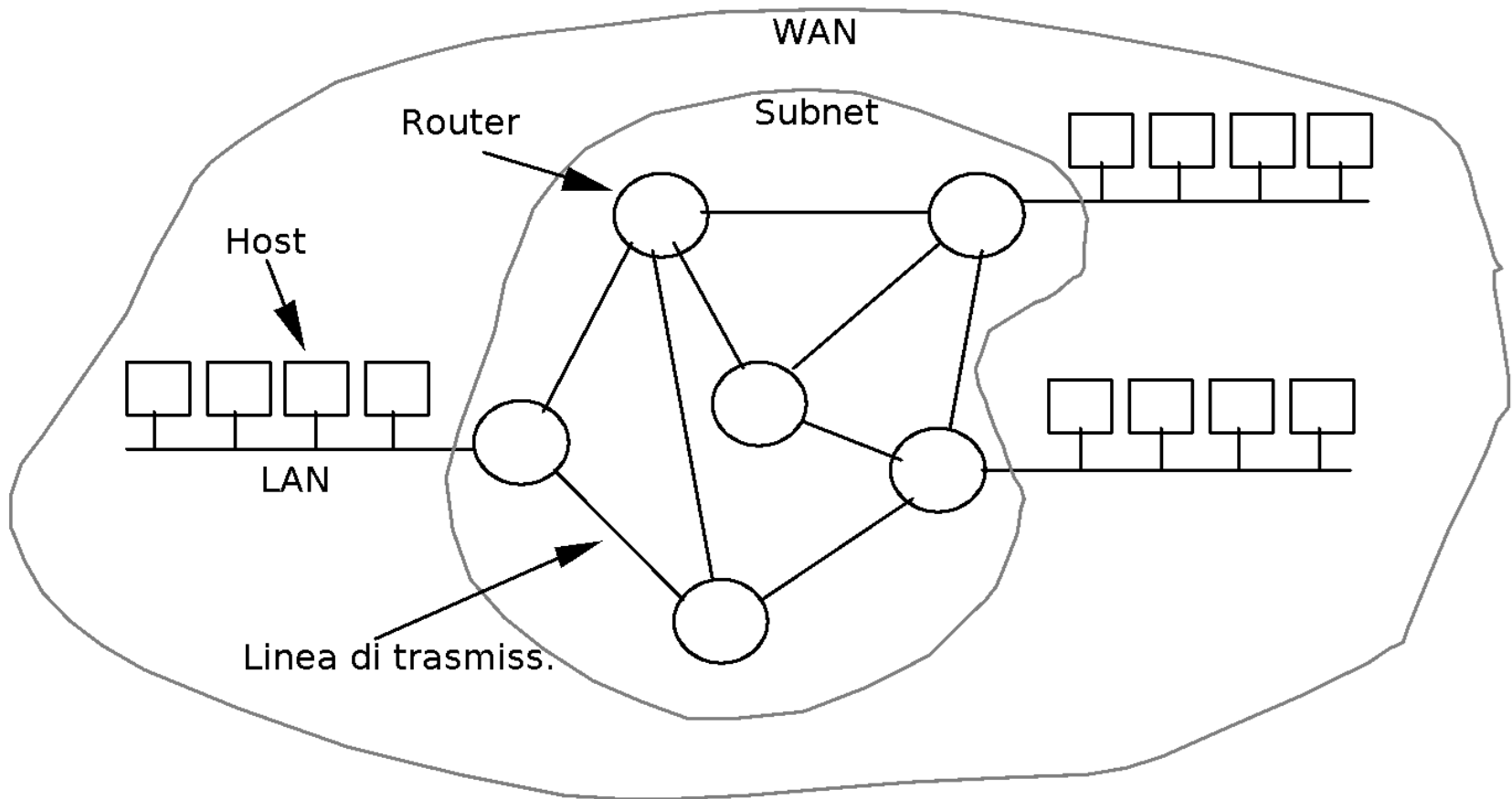
- La dimensione non può andare oltre un certo limite
 - ✓ è noto a priori il tempo di trasmissione nel caso peggiore
 - Questa conoscenza permette di utilizzare delle tecniche particolari per la gestione del canale di comunicazione

- Le LAN sono quasi sempre (combinazioni di) reti broadcast

Reti WAN

- WAN è acronimo di **Wide Area Network (WAN)**
- Si estendono a livello di una nazione, di un continente o dell'intero pianeta
- La più famosa rete WAN è Internet (con la *i* maiuscola)
- Spesso una WAN connette più LAN in modo **punto a punto**
- In alcuni testi si può trovare anche il termine MAN
- MAN è acronimo di *Metropolitan Area Network*
- Indica le reti che si estendono all'interno di una città
- Il concetto di MAN si collega al concetto emergente di *smart cities*

Classificazione per dimensione



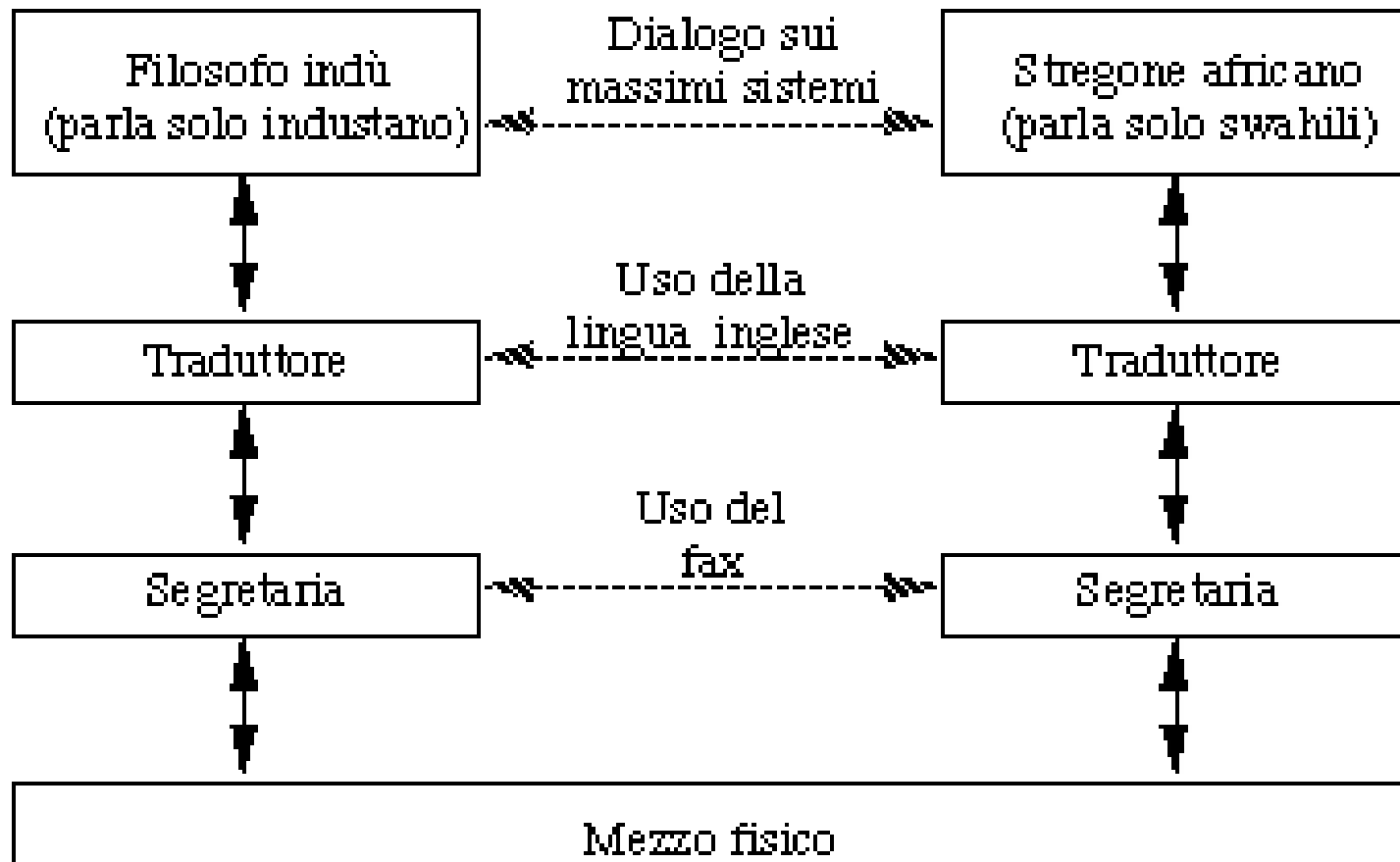
Reti: principi di funzionamento

- Le reti sono state progettate “a strati”, meglio chiamati **livelli (layer)**.
- Per capire il perché si consideri il seguente esempio.

Un grande filosofo indù che parla solo il particolarissimo dialetto industano vuole dialogare sui massimi sistemi con un grande stregone africano che parla solo il particolarissimo dialetto swahili. Affinché il dialogo abbia luogo è necessario trovare un interprete industano/swahili che però data la particolarità delle lingue non si riesce a trovare. E' possibile tuttavia reperire due interpreti: uno industano/inglese e l'altro swahili/inglese. Ecco pertanto che il dialogo tra i due interpreti potrebbe avere luogo. I due interpreti sono geograficamente lontani e decidono di scambiarsi le informazioni via fax coinvolgendo ciascuno una segretaria. Il dialogo tra il filosofo e lo stregone avviene pertanto come in figura:

(riadattato da A. Tanenbaum, *Computer Networks*, 5th Ed., Prentice Hall, 2011)

Reti: principi di funzionamento - II



Livelli, protocolli e servizi

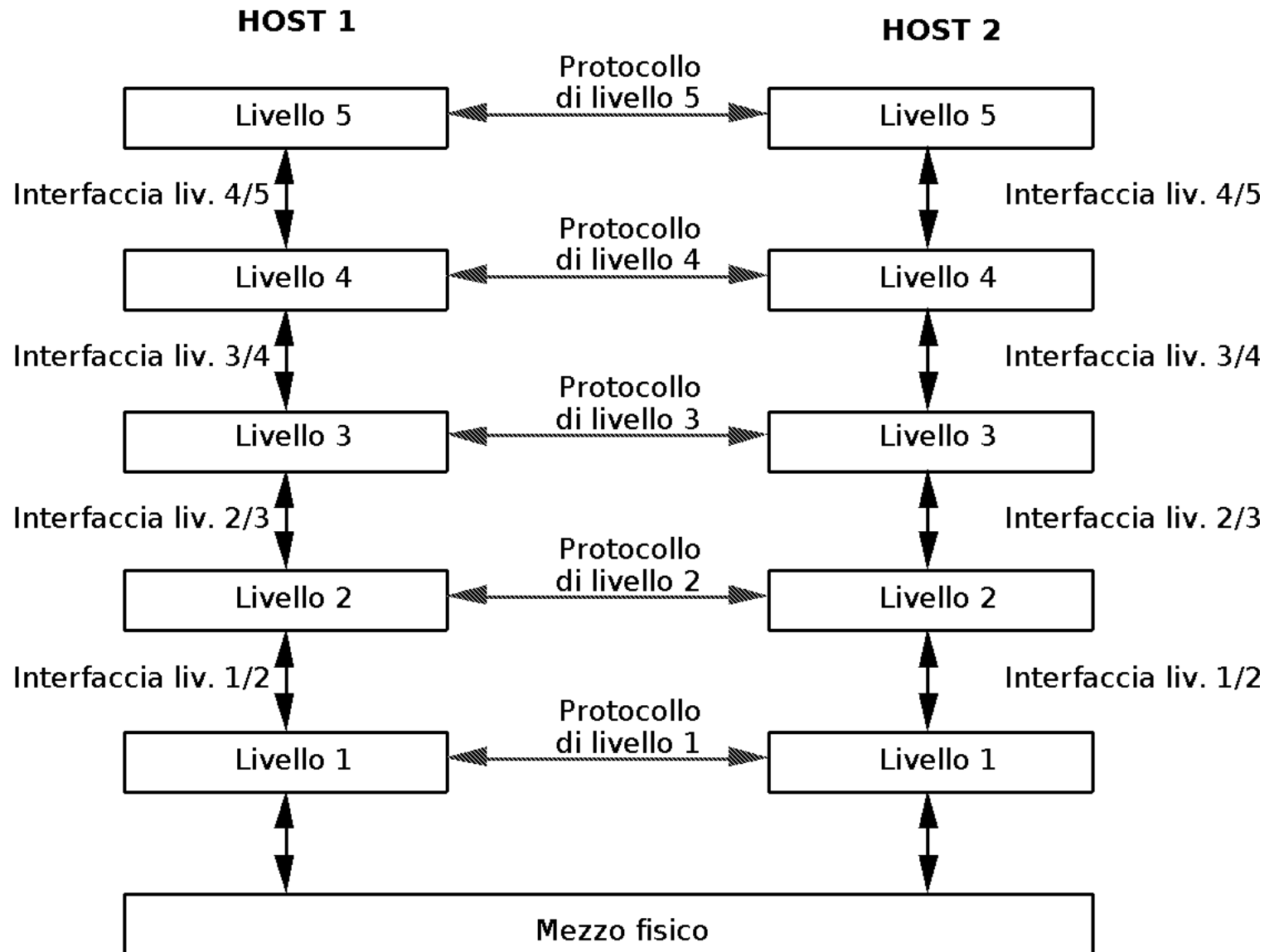
- Le reti sono state organizzate a **livelli**, ciascuno costruito sopra il precedente
- Realizzare la comunicazione tra due calcolatori è un problema complesso
 - ✓ E' necessario scomporre il problema in sottoproblemi
- Lo scopo di un **livello** è offrire un **servizio** ai livelli più alti
- Un livello nasconde i dettagli su come tali servizi siano implementati
- Idealmente, il livello n su un host comunica col livello n di un altro host
- Le **regole** e le **convenzioni** della conversazione sono il **protocollo** di livello n .

Livelli, protocolli e servizi - II

- Tornando all'analogia umana, da un punto di vista **logico**:
 - ✓ il filosofo “parla” con lo stregone (e viceversa)
 - ✓ l'interprete “parla” con l'altro interprete
- Nella **realtà**:
 - ✓ Il filosofo si appoggia al servizio offertogli dal suo interprete
 - ✓ Lo stregone si appoggia al servizio offertogli dal suo interprete
- E' importante capire che:
 - ✓ Lo stregone non parla con la segretaria
 - ✓ Allo stregone non importa sapere che c'è una segretaria (che usa un fax)
 - ✓ La segretaria non sa che sta spedendo il messaggio di uno stregone

Livelli, protocolli e servizi - III

- Il dialogo fra due **entità di pari livello n (peer entity)** avviene:
 - ✓ Sfruttando i servizi offerti dal livello $n-1$
 - ✓ Fra i livelli n ed $n-1$ è definita **un'interfaccia**, ovvero l'insieme delle operazioni che possono essere richieste al livello sottostante
- Le peer entity concettualmente comunicano «in orizzontale» fra loro
- In realtà comunicano ciascuna col livello sottostante
- Oltre ai dati vengono scambiate anche informazioni di controllo
- Al di sotto del livello più basso c'è il mezzo fisico (un cavo o un antenna)
- Attraverso il mezzo fisico i dati sono fisicamente trasferiti tra gli host



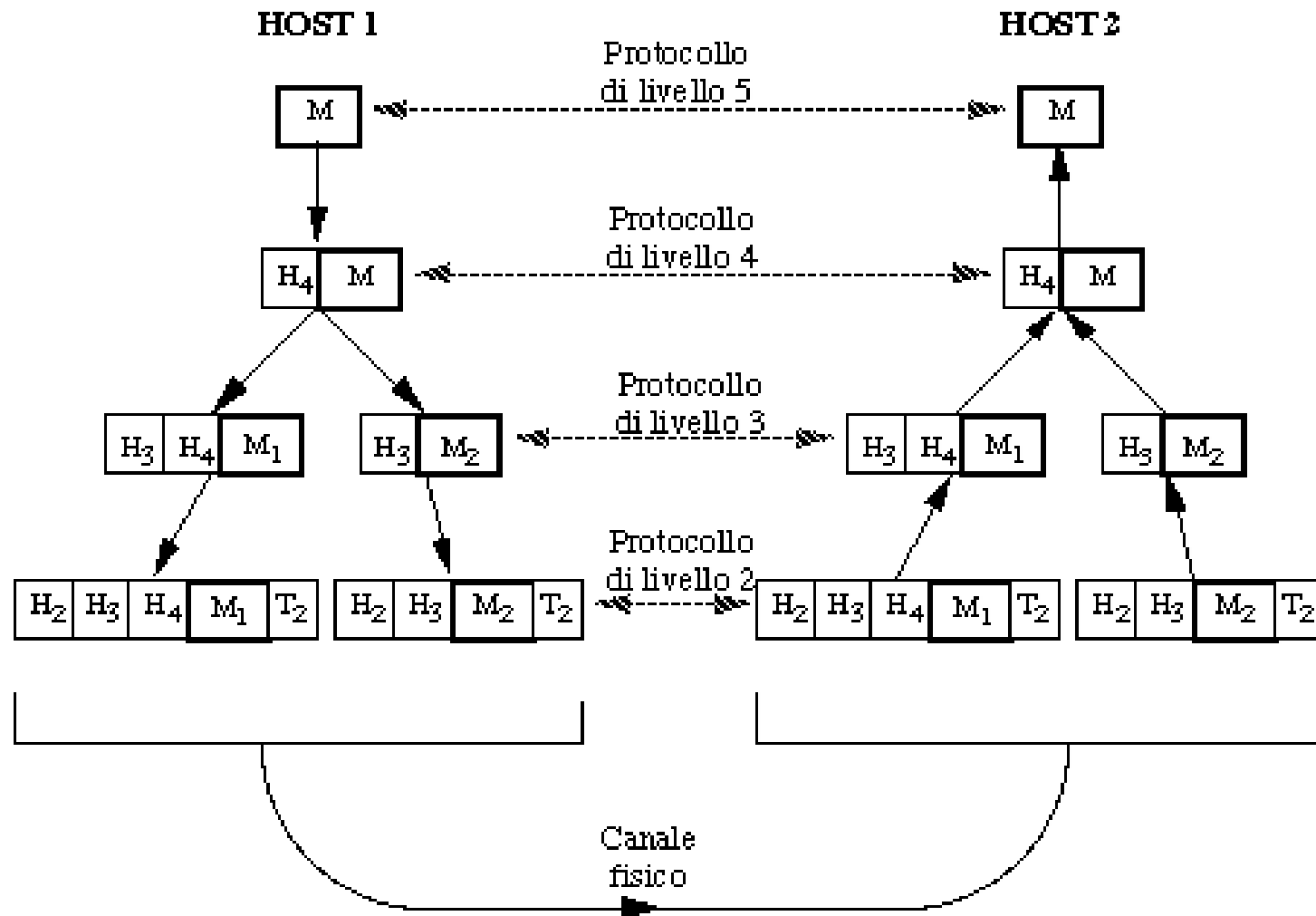
Comunicazione tra livelli

- Una situazione più dettagliata è la seguente (vedi figura successiva):
 - ✓ il programma applicativo (livello 5) deve mandare un messaggio M alla sua peer entity;
 - ✓ il livello 5 consegna M al livello 4 per la trasmissione;
 - ✓ il livello 4 aggiunge un suo **header** in testa al messaggio (talvolta si dice che il messaggio è inserito nella **busta** di livello 4)
L' header contiene informazioni di controllo, come ad esempio il numero di sequenza del messaggio, la dimensione, ecc...
 - ✓ il livello 4 consegna il risultato al livello 3;
 - ✓ ...

Comunicazione tra livelli - II

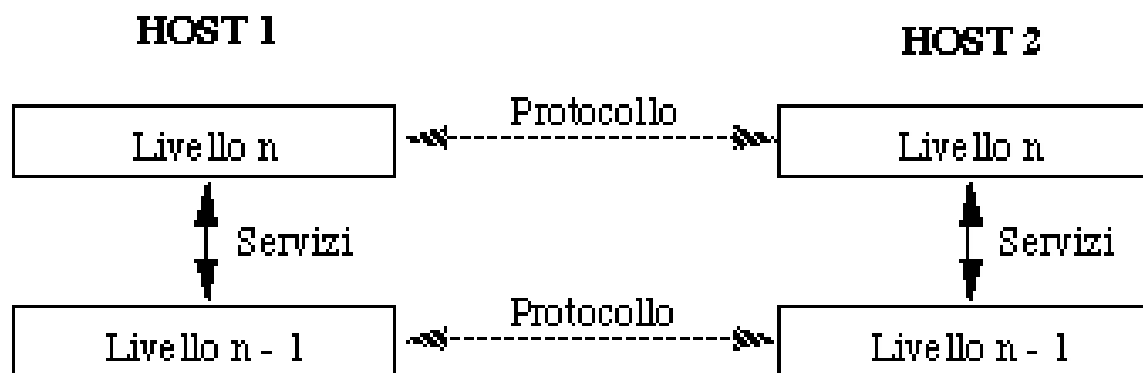
- ✓ ...
- ✓ il livello 3 può trovarsi nella necessità di **frammentare i dati** da trasmettere in unità più piccole (**pacchetti**) a ciascuna delle quali aggiunge il suo header
- ✓ il livello 3 passa i pacchetti al livello 2
- ✓ il livello 2 aggiunge ad ogni pacchetto il proprio header (e magari anche un **trailer**) e lo spedisce sul livello 1 (canale fisico);
- ✓ nella macchina di destinazione i pacchetti fanno il percorso inverso, con ogni livello che elimina (elaborandoli) l'header ed il trailer di propria competenza, e passa il resto al livello superiore.

Comunicazione tra livelli: messaggi, header, trailer



Ancora su protocolli e servizi

- **Servizio**: insieme di operazioni primitive che un livello offre al livello superiore. Come tali operazioni sono svolte non riguarda il livello superiore
- **Protocollo**: insieme di regole che governano il formato ed il significato delle informazioni che le peer entity si scambiano fra loro
- Un livello usa il protocollo (il “come”) per implementare il servizio (il “cosa”)

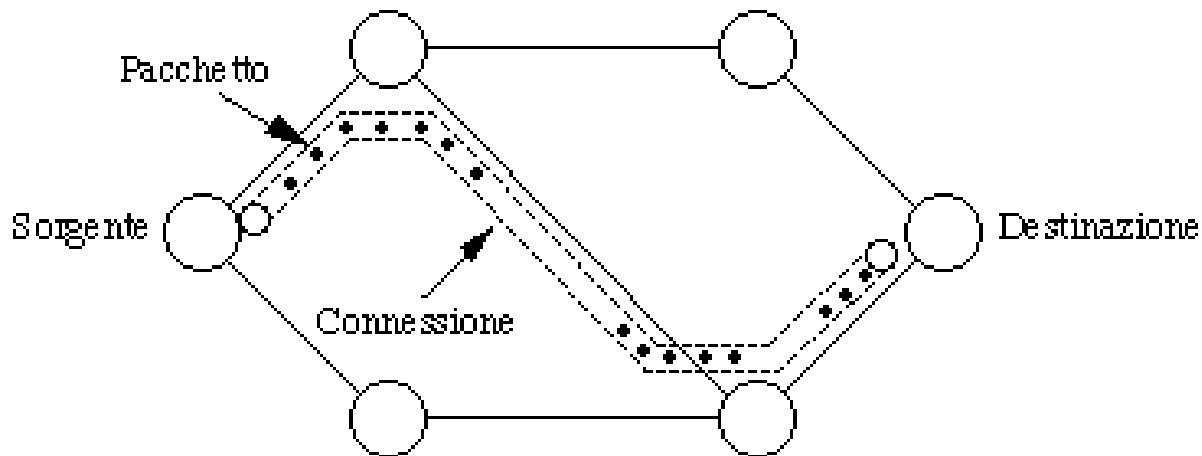


Tipologia di servizi

- Il servizio offerto da un livello al livello superiore può essere:
 - ✓ **Connection oriented** (o **connesso**):
 - sono modellati **secondo il sistema telefonico**
 - si alza il telefono, si chiama, si parla e poi si riattacca
 - ✓ **Connectionless**:
 - sono modellati **secondo il sistema postale**
 - ogni pacco viaggia indipendentemente dagli altri
 - arriva quando arriva
 - pacchi con uguale mittente e destinatario possono avere percorsi diversi

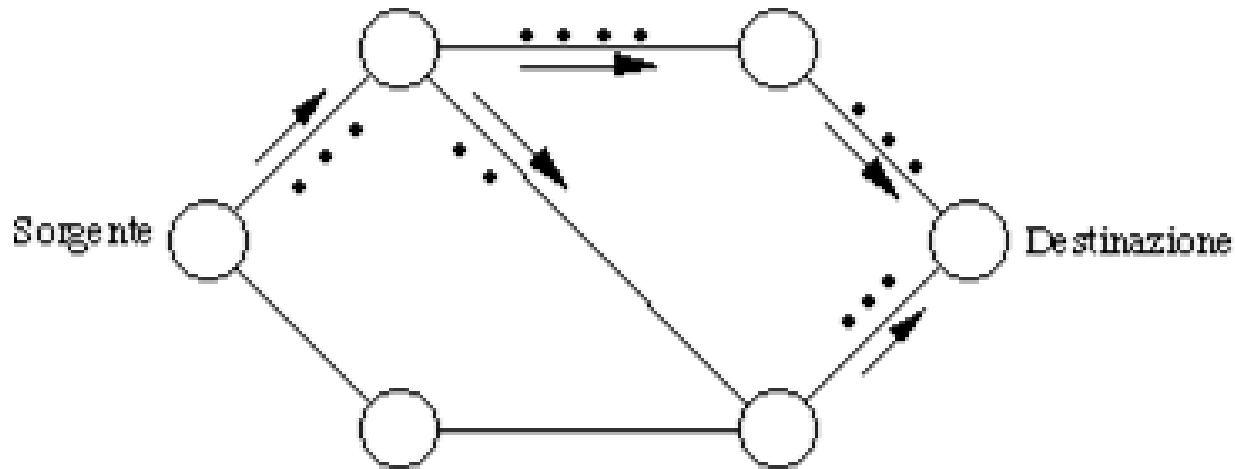
Servizi *connection-oriented*

- Analogamente, un servizio *connection-oriented* si sviluppa in 3 fasi:
 - ✓ si stabilisce una connessione, cioè si crea con opportuni mezzi un "canale di comunicazione" fra la sorgente e la destinazione.
 - ✓ la connessione, una volta stabilita, agisce come un **"tubo digitale"**
 - i dati trasmessi arrivano nello **stesso ordine** in cui sono partiti
 - ✓ si rilascia la connessione



Servizi *connectionless*

- In un servizio *connectionless*.
 - ✓ i pacchetti viaggiano indipendentemente gli uni dagli altri
 - ✓ possono prendere strade diverse
 - ✓ possono arrivare in **ordine diverso** da quello di partenza



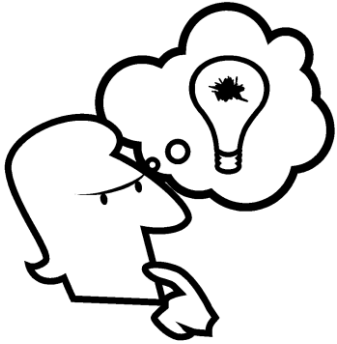
Tipologia di servizi - II

- E' necessario ricordare che:
 - ✓ Le reti di calcolatori possono generare **messaggi duplicati**
 - ✓ (si pensi all'analogia con il sistema telefonico in cui esiste l'eco)
- Il servizio offerto da un livello al livello superiore può essere:
 - ✓ **Affidabile** (inglese: *reliable*):
 - è garantito che non vengono mai persi dati ("arriva tutto")
 - è garantito che non vengono generati duplicati
 - ✓ **Non affidabile** (inglese: *unreliable*)
 - non è garantita la consegna di tutti i dati
 - non è garantito che non vengano generati duplicati

Classificazione dei servizi

- Diventano interessanti le combinazioni dei tipi precedenti, ovvero:
 - ✓ Servizi **connessi** e **affidabili**, nei quali i dati trasmessi
 - arrivano tutti
 - arrivano in ordine
 - arrivano senza duplicati
 - (c'è una vaga analogia con la raccomandata con ricevuta di ritorno)
 - ✓ Servizi **connectionless** e **non affidabili**, per i quali
 - non è garantito che arrivi tutto
 - non è garantito che ciò che arriva arrivi in ordine
 - non è garantito che non si generino duplicati
 - (analogia con il servizio postale più economico)

Classificazione dei servizi



**... ma di che utilità è un servizio
connectionless e non affidabile?**

- Una raccomandata con ricevuta di ritorno costa più della posta ordinaria...
- Nelle reti ciò che "costa" è l'occupazione del canale trasmissivo
- Per offrire un servizio affidabile e connesso servono molti dati di controllo
- Molte applicazioni possono tollerare la perdita di porzioni del messaggio
- Queste applicazioni spesso vogliono trasmettere molti dati in breve tempo
 - ✓ es. videostreaming

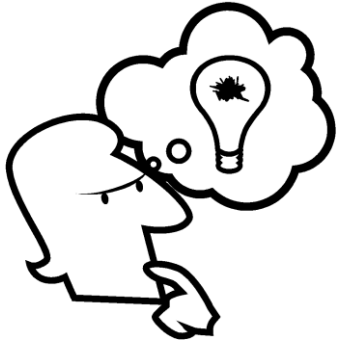
Progettazione di un protocollo/servizio di rete

- Decisioni di progetto vanno prese, nei vari livelli, su diverse problematiche:
 - ✓ Meccanismi di identificazione di mittente e destinatario (**indirizzamento**)
 - ✓ Regole per il **trasferimento dati** (livelli bassi):
 - in una sola direzione (**simplex connection**)
 - in due direzioni non contemporaneamente (**half-duplex connection**)
 - in due direzioni contemporaneamente (**full-duplex connection**)
 - ✓ Meccanismi per il controllo degli **errori di trasmissione**; è possibile:
 - rilevarli oppure no
 - correggerli oppure no
 - avvertire il mittente oppure no
 - ✓ ...

Progettazione di un protocollo/servizio di rete - II

- ✓ ...
- ✓ Mantenimento (o ricostruzione) dell'**ordine originario** dei dati
- ✓ Meccanismi per regolare le **velocità** di sorgente e destinazione
- ✓ Decisioni sulla **dimensione** (minima o massima) dei messaggi da inviare
 - Decisioni su come eventualmente frammentarli
- ✓ Meccanismi di **multiplexing** di varie "conversazioni" su di un'unica connessione (se stabilire la connessione è costoso)
- ✓ Meccanismi di **routing** dei messaggi se esistono più strade alternative, ed eventualmente di suddivisione di una "conversazione" su più connessioni contemporaneamente (per aumentare la velocità di trasferimento dei dati)

Architetture di rete

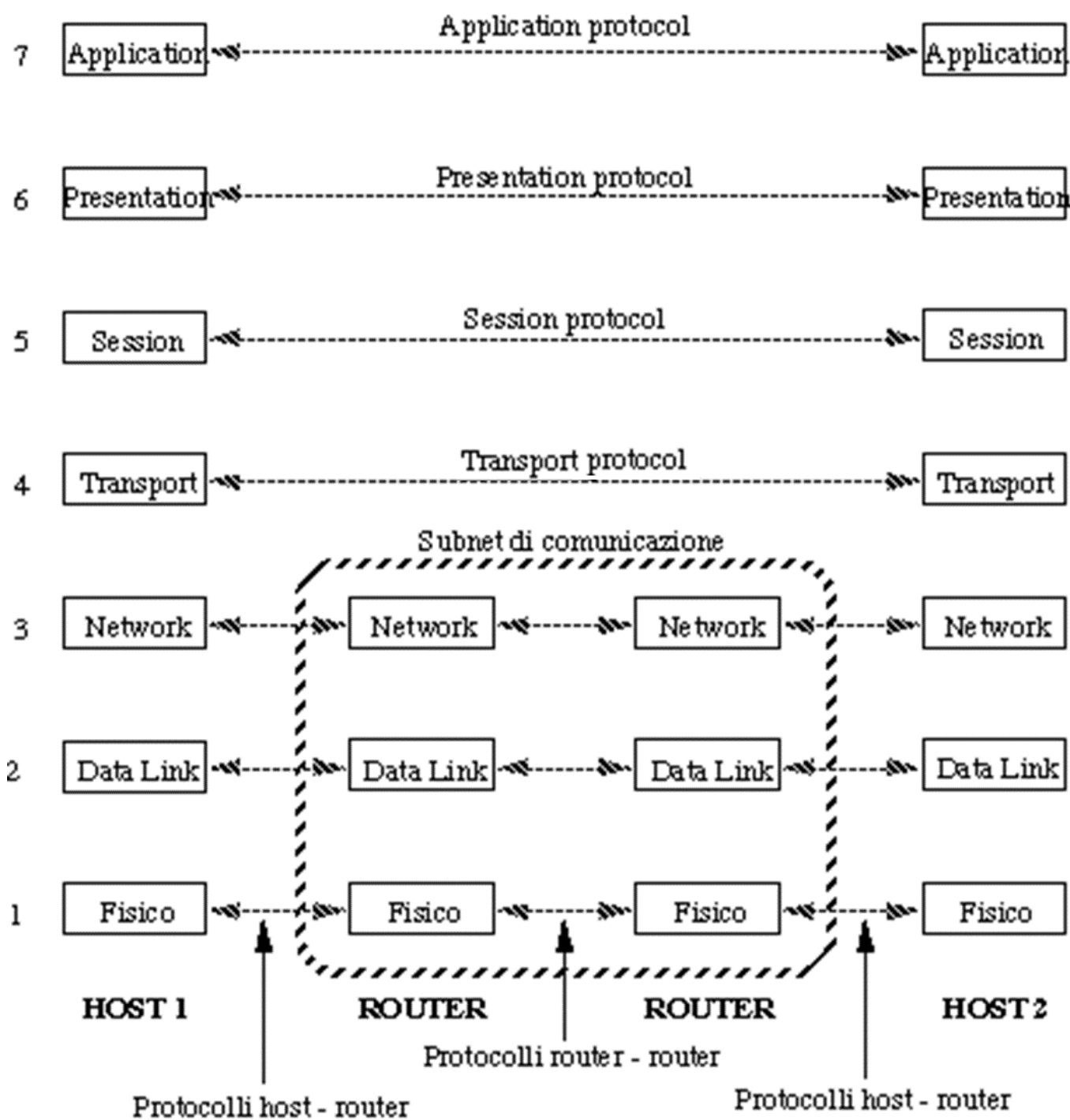


... ma nella pratica quanti e quali livelli esistono?

- Nella storia delle reti è accaduto che:
 - ✓ E' stato proposto un modello chiamato ISO/OSI ma non ha preso piede
 - ✓ Si utilizza l'architettura TCP/IP che ha dei legami con ISO/OSI
- Per essere precisi, ISO/OSI è un modello di rete («teorico» e poco «pratico»)
- TCP/IP è un'architettura di rete (è «pratica» ma non è frutto di un modello)
- Partiamo da ISO/OSI...

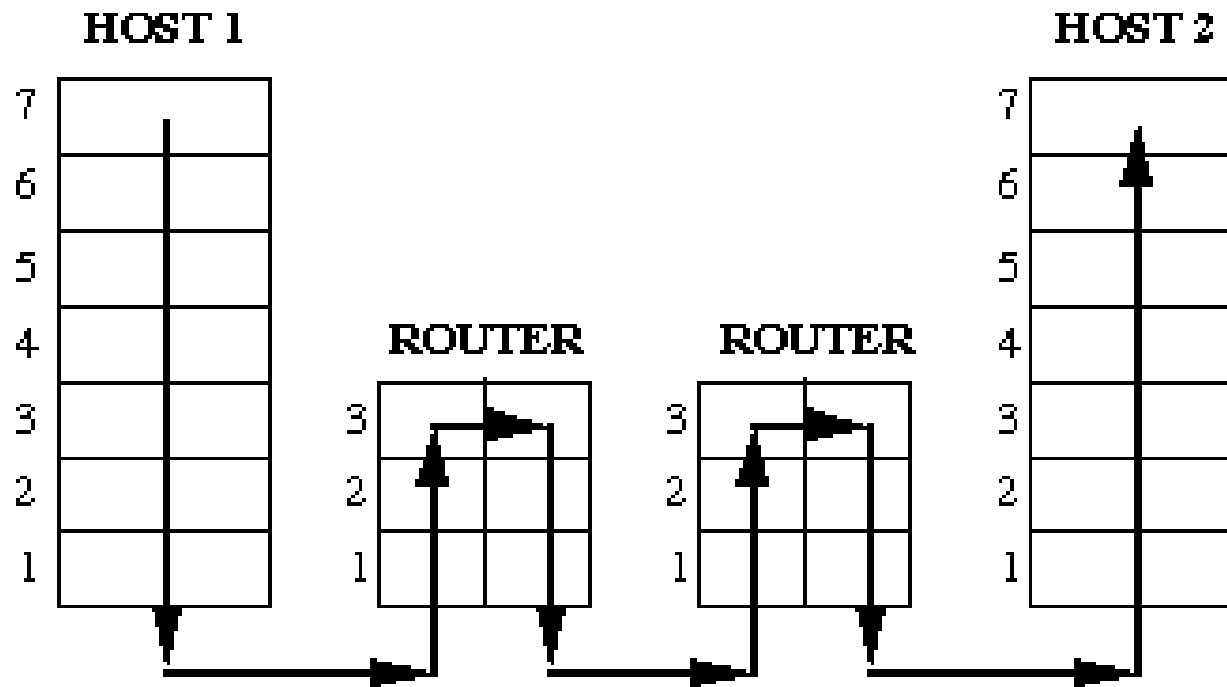
Il modello ISO/OSI

- OSI sta per *Open Systems Interconnection*
- E' frutto del lavoro della ISO (*International Standard Organization*)
- Ha lo scopo di:
 - ✓ fornire uno standard per la connessione di sistemi aperti
 - ✓ fornire un modello rispetto a cui confrontare le varie architetture di rete
- Esso **non include la definizione di protocolli** specifici
- I protocolli sono stati definiti successivamente in documenti separati
- In realtà nessuno dei protocolli proposti da ISO/OSI è usato nella pratica
- Il risultato è un modello basato su **7 livelli**:



Il modello ISO/OSI

- Si può rappresentare la situazione anche nel seguente modo:



- Vediamo ora cosa dovrebbe fare ciascun livello secondo ISO/OSI

Livello 1 - Fisico

- Si occupa della **trasmissione di bit "grezzi" su un canale** di comunicazione
- Gli aspetti di progetto riguardano prevalentemente le caratteristiche meccaniche, elettriche del mezzo fisico (ad esempio cavo o antenna)
- E' a questo livello che ad esempio è necessario determinare:
 - ✓ tensioni elettriche scelte per rappresentare 0 ed 1
 - ✓ durata (in microsecondi) di un bit
 - ✓ trasmissione simultanea in due direzioni oppure no
 - ✓ forma dei connettori

Livello 2 – Data Link

- Si preoccupa che un mezzo fisico trasmissivo appaia, al livello superiore, come una linea di trasmissione esente da errori di trasmissione non rilevati.
- Gli aspetti di progetto riguardano prevalentemente garantire che se parte un 1, arrivi effettivamente un 1 e non uno zero, visto che ogni mezzo trasmissivo ha un suo grado di affidabilità (interferenze, errori, ecc...)
- E' a questo livello che ad esempio è necessario determinare:
 - ✓ quanti bit «al colpo» trasmettere (grandezza del frame)
 - ✓ se, quanti e quali bit ridondanti di controllo aggiungere
 - ✓ lasciare che il ricevente possa solo rilevare un errore o anche correggerlo (e fino che a punto correggere un errore)

Livello 3 – Network

- Si preoccupa dell'instradamento nella subnet di comunicazione
- Il suo ruolo è permettere ad un host di iniettare pacchetti in una qualunque rete
- Il livello poi si deve occupare di farli viaggiare fino a destinazione

- E' a questo livello che ad esempio è necessario determinare:
 - ✓ come identificare i nodi (**indirizzi**)
 - ✓ come gestire il **routing** (qual è la strada migliore)

- E' il livello di fatto più importante nelle reti di calcolatori
- Nella pratica si è adottata una sola strategia (offrire un servizio *connectionless*)

Livello 4 – Trasporto

- E' il primo livello **end-to-end**
- Ha delle incombenze simili al livello 2 ma non per i singoli bit ma per i pacchetti
- I pacchetti possono prendere strade diverse nella subnet di comunicazione
- E' a questo livello che ad esempio è necessario determinare:
 - ✓ quanti pacchetti trasmettere (controllo del flusso)
 - ✓ se dover creare un canale per un servizio *connection-oriented*
 - ✓ se richiedere la ritrasmissione in caso di servizio affidabile
 - ✓ come gestire diverse applicazioni dei livelli superiori che contemporaneamente vogliono fare accesso alla rete

Livello 5 – Sessione

- Non era chiaro secondo OSI di cosa si dovesse occupare questo livello
- L'idea era di autorizzare le due parti, a turno, alla trasmissione
- **Non ha avuto molto successo** nella pratica

Livello 6 – Presentazione

- Non era chiaro secondo OSI di cosa si dovesse occupare questo livello
- L'idea era di gestire a questo livello **conversioni di formato**
- Il trasmittente potrebbe scegliere un «formato» diverso da quello del ricevente
- E' un problema già incontrato nella lezione sulla codifica dell'informazione:
 - ✓ Rappresentiamo i caratteri con ASCII o UNICODE?
 - ✓ Rappresentiamo gli interi in complemento a 2 o modulo e segno?
- Questo livello doveva occuparsi di questi aspetti
- **Non ha avuto molto successo** nella pratica

Livello 7 – Applicazione

- E' il livello dei servizi finali all'utente
- Qui sono definiti i protocolli che agiscono da interfaccia "uomo/calcolatore"
- Ad esempio:
 - ✓ terminale virtuale
 - ✓ trasferimento file
 - ✓ posta elettronica
 - ✓ web

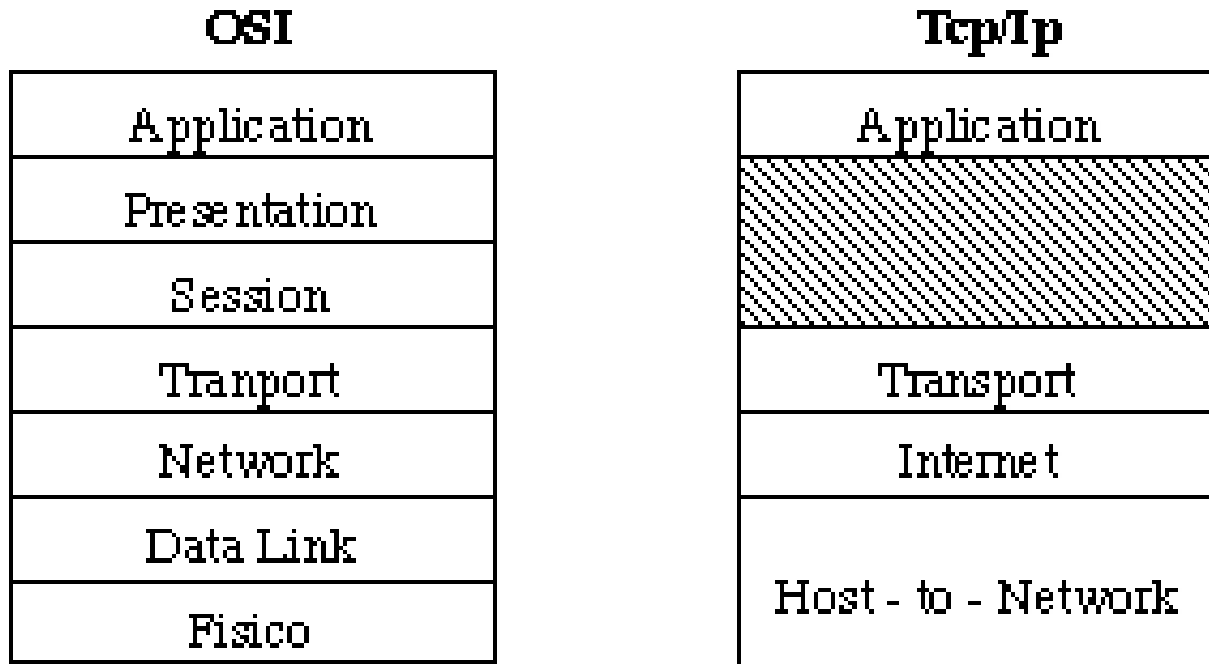
L'architettura TCP/IP

- L'architettura di rete usata nella pratica non segue alla lettera ISO/OSI
- Non ha un nome alternativo, si parla di **architettura TCP/IP**
- Prende il nome dai suoi due protocolli principali

- Essa non è un modello nel senso stretto del termine
- Include i protocolli effettivi, specificati per mezzo di documenti detti RFC (Request For Comments)

L'architettura TCP/IP

- L'analogia con ISO/OSI è la seguente:



- I livelli 5 e 6 sono spariti e si parla genericamente di livello "host-to-network"

I protocolli di TCP/IP

- TCP/IP nasce anche con la **definizione dei protocolli**
- In particolare quelli di livello trasporto e internet danno il nome all'architettura

Application	Telnet	Ftp	Smtp	Http	Nntp	ecc.
Transport	Tcp		Udp			
Internet	IP					
Host - to - Network	Vari standard per LAN, WAN e MAN					

- I protocolli del livello applicativo riportati nello schema sono solo alcuni esempi

ISO/OSI vs TCP/IP

- **Somiglianze:**

- ✓ basati entrambi sul concetto di pila (o stack) di protocolli indipendenti
- ✓ funzionalità simili in entrambi per i vari livelli

- **Differenze:**

- ✓ OSI nasce come modello di riferimento e rimane standard *de jure* (utile per discussioni generali), i protocolli vengono solo successivamente
- ✓ TCP/IP nasce coi protocolli, il modello di riferimento viene a posteriori ma diventa standard *de facto*

Riassunto dei concetti chiave

- Broadcast e punto a punto
- LAN e WAN
- Livello
- Servizio
- Protocollo
- Interfaccia
- ISO/OSI
- TCP/IP