



# Networking



Università degli Studi di Milano

# Introduzione

- ▶ Nel 20° secolo si sono diffusi
  - ▶ Sistema telefonico a livello mondiale
  - ▶ Radio e televisione
  - ▶ Computer e internet
  - ▶ Satelliti per le comunicazioni
  - ▶ Telefonia mobile
- ▶ Nel 21° secolo
  - ▶ Smartphone e app
  - ▶ Applicazioni web
  - ▶ Cloud computing
  - ▶ Big data
  - ▶ IoT



# Storia delle reti

- ▶ Mainframe - terminali
  - ▶ Potenza di calcolo in un unico elaboratore
  - ▶ Accesso tramite terminale
- ▶ Rete di elaboratori (computer network)
  - ▶ Elaboratori autonomi e interconnessi
- ▶ Internet
  - ▶ Rete di reti
  - ▶ Topologia distribuita e decentralizzata



# Reti di calcolatori

- ▶ Usati dalle organizzazioni per
  - ▶ Condivisione di risorse
  - ▶ Affidabilità
  - ▶ Diminuzione dei costi
  - ▶ Scalabilità
  - ▶ Comunicazione fra persone
- ▶ Usato dagli utenti per
  - ▶ Accesso a informazioni remote
  - ▶ Comunicazione tra utenti
  - ▶ Divertimento e social networking



# Reti di calcolatori

- ▶ «Esigenza fondamentale di ogni applicazione che coinvolge due o più elaboratori è quella di fornire un adeguato sistema di comunicazione dati» (Fred Halsall)
- ▶ La gamma di sistemi di comunicazione utilizzabili è molto ampia, ciascuno riferito ad un diverso dominio applicativo (ad es., trasferimento dati tra computer di una stessa stanza vs computer in luoghi differenti)



# Reti di calcolatori

- ▶ I sistemi di comunicazione dipendono
  - ▶ Dalla natura dell'applicazione
  - ▶ Numero di calcolatori coinvolti
  - ▶ La loro distanza fisica
- ▶ Considerando solo 2 calcolatori coinvolti
  - ▶ Stessa stanza: singolo collegamento punto punto
  - ▶ Luoghi diversi: devo usare linee pubbliche (PSTN - Public Switched Telephone Network)



# Reti di elaboratori – Scala dimensionale

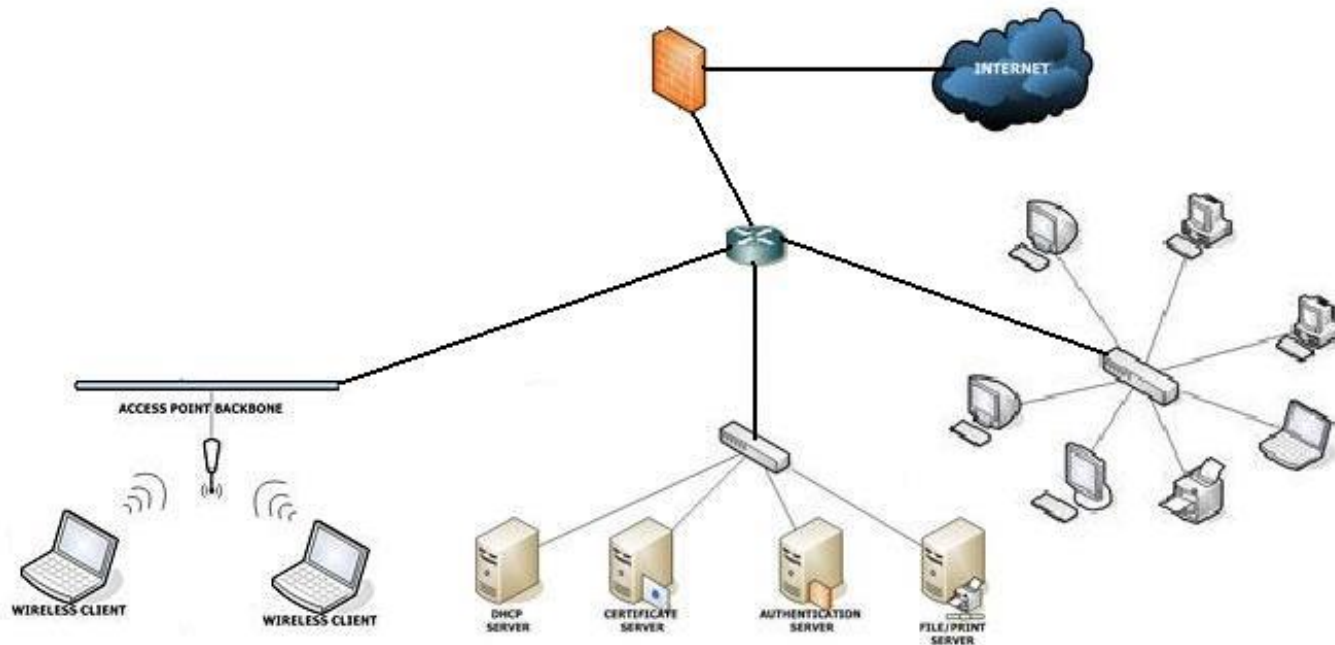
- ▶ Reti locali (LAN)
- ▶ Reti metropolitane (MAN)
- ▶ Reti geografiche (WAN)

Distanza fra processori	Ambito	Tipo di rete
10 m.	Stanza	Rete locale
100 m.	Edificio	Rete locale
1 km.	Campus	Rete locale
10 km.	Città	Rete metropolitana
100 km.	Nazione	Rete geografica
1000 km.	Continente	Rete geografica
10.000 km.	Pianeta	Internet (Rete geografica)



# Reti locali (LAN)

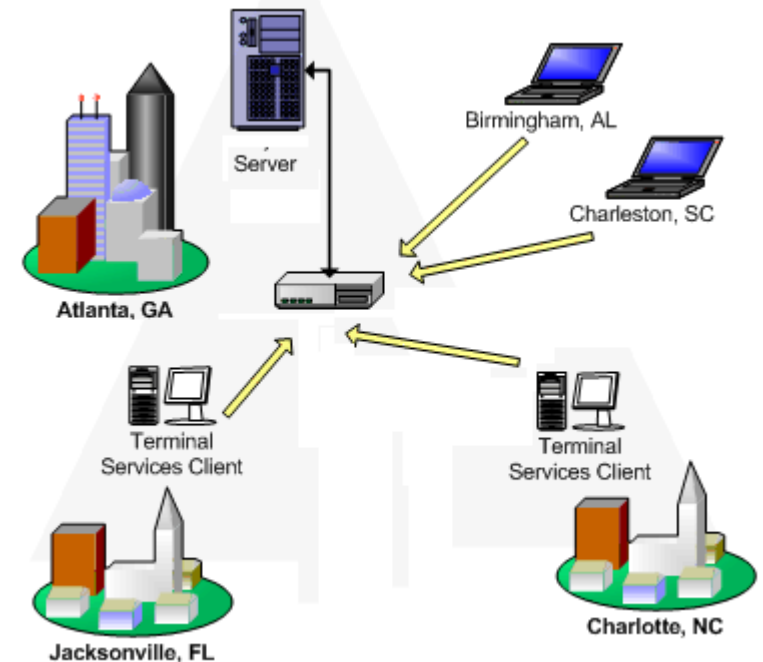
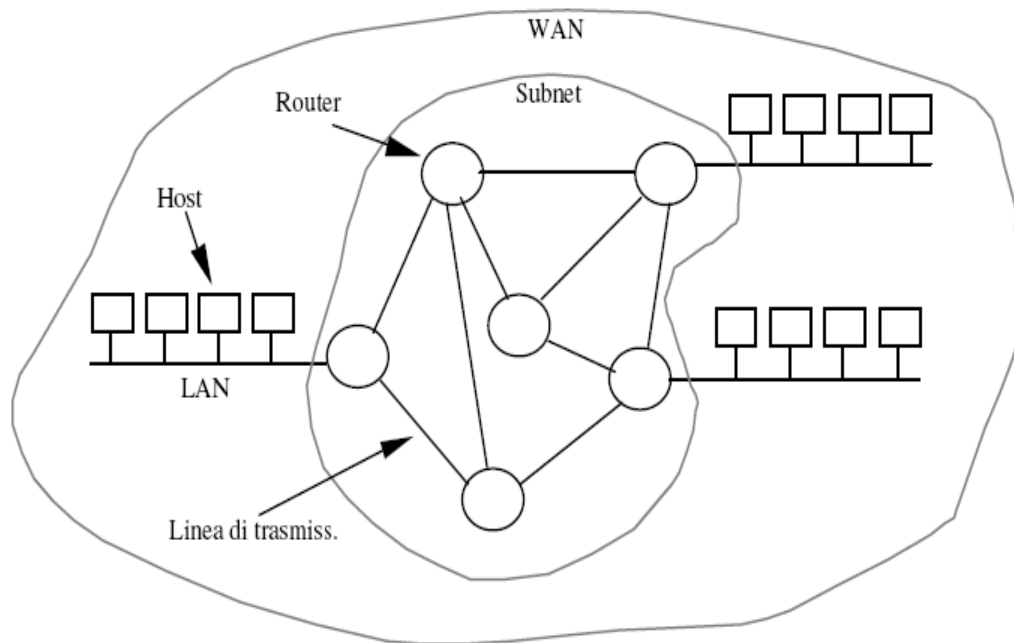
- ▶ Gestite da organizzazioni singole
- ▶ Qualche km di estensione
- ▶ Non risiedono su suolo pubblico (singolo edificio, campus)
- ▶ Usate per connettere pc o workstation





# Reti geografiche (WAN)

- ▶ Estesa a livello di nazione, continente, pianeta
- ▶ Costituita da
  - ▶ Insieme di calcolatori
  - ▶ Subnet di comunicazione (router e linee trasmissive)



# Internet

- ▶ Rete di reti (LAN, MAN, WAN)
- ▶ Rete distribuita, simile ad una WAN...
- ▶ ...ma è diversa
  - ▶ Connette reti eterogenee
  - ▶ Ha bisogno di attrezzature speciali (gateway)



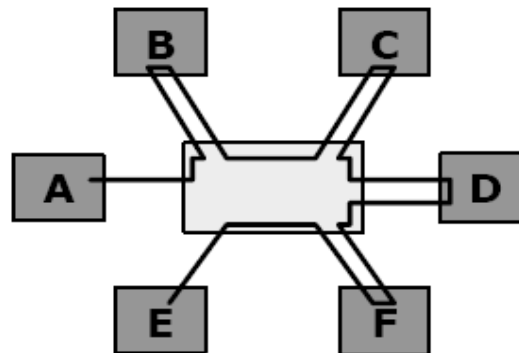
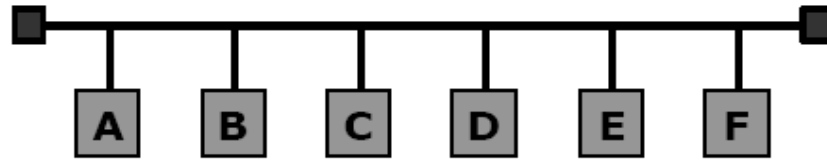
# Topologie

- ▶ Ogni tipologia di rete può essere fisicamente realizzata sfruttando topologie differenti quali
  - ▶ Bus
  - ▶ Stella
  - ▶ Anello



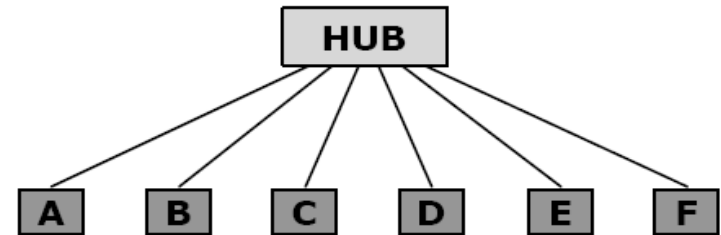
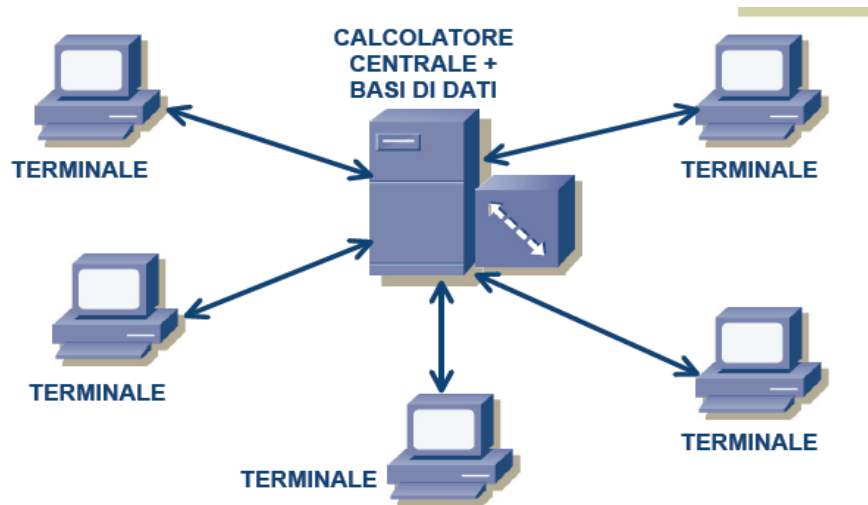
# Topologie

- Topologia logica a Bus, cablata a Stella



# Topologie

## ► Rete centralizzata vs topologia a stella



# Standard

- ▶ La comunicazione tra nodi differenti e, possibilmente, basati su piattaforme hardware e/o software eterogenee necessita di standard
- ▶ Storicamente si parla di standard di un produttore: scambio di informazioni possibile solo per sistemi della stessa marca (sistemi chiusi)
- ▶ Standard definiti da enti internazionali (livello di trasmissione dati)
  - ▶ Serie V per DTE (Data Terminal Equipment) a un modem connesso a rete telefonica
  - ▶ Serie I per le connessioni di DTE a linee ISDN
  - ▶ Serie X per DTE connesso a rete pubblica per la trasmissione dati
- ▶ Standard definiti dagli enti e dai fornitori di servizi detti standard di alto livello
  - ▶ Riguardano il formato e il controllo dello scambio di informazioni



# Standard

## ▶ Vantaggi

- ▶ Assicurano un vasto mercato per i dispositivi e il software
- ▶ Consentono a prodotti provenienti da fornitori differenti di comunicare

## ▶ Svantaggi

- ▶ Tecnologia congelata
- ▶ Possono esserci più standard per una stessa funzione



# Standard

- ▶ Fin dagli albori, l'informatica, conosce due tipi di standard
  - ▶ Standard de iure
  - ▶ Standard de facto





# Standard ISO-OSI

- ▶ Verso la metà degli anni 70 l'industria informatica si rese conto dei vantaggi dei sistemi aperti
- ▶ Venne prodotto dalla International Standard Organization (ISO) lo standard che regola la struttura globale del sottosistema completo di comunicazione
  - ▶ Noto come Modello di Riferimento ISO per l'interconnessione di sistemi aperti (OSI - Open System Interconnection)



# Standard ISO-OSI

- ▶ Un sistema di comunicazione è decisamente complesso
  - ▶ Un'implementazione non strutturata basata su «un singolo programma» non è facile da testare e modificare
- ▶ Per questo motivo ISO ha adottato un modello a strati
- ▶ Il sistema di comunicazione viene diviso a strati ciascuno dei quali esegue una funzione predefinita



# Standard ISO-OSI

- ▶ Strati ISO/OSI possono essere separati in due categorie
  - ▶ Funzioni dipendenti dalla rete (**Media layers**)  
(Physical, Data link, Network)
  - ▶ Funzioni orientate all'applicazione (**Host layers**)  
(Transport, Session, Presentation, Application)

## ISO/OSI

Application
Presentation
Session
Transport
Network
Datalink
Physical



# Standard ISO-OSI

- ▶ La funzione di ogni strato è codificata da una serie di regole e convenzioni usate per comunicare con lo strato remoto corrispondente (**protocollo**)
- ▶ Ogni strato fornisce dei servizi allo strato immediatamente superiore e utilizza inoltre dei servizi forniti dallo strato immediatamente inferiore



# Protocollo

- ▶ La comunicazione tra entità richiede cooperazione, ossia collaborazione per il conseguimento di uno scopo comune
  - ▶ Comunicazioni regolate mediante protocolli
- ▶ **Protocollo:** insieme di regole e convenzioni seguite da entità, dislocate su nodi distinti, che intendono comunicare per svolgere un compito comune
  - ▶ Tali regole hanno l'obiettivo di assicurare una cooperazione efficiente e affidabile per la comunicazione tra nodi, l'effettuazione di servizi considerando le caratteristiche tipiche di un sistema distribuito (banda di trasmissione limitata, ritardi variabili, errori nella comunicazione, ...)



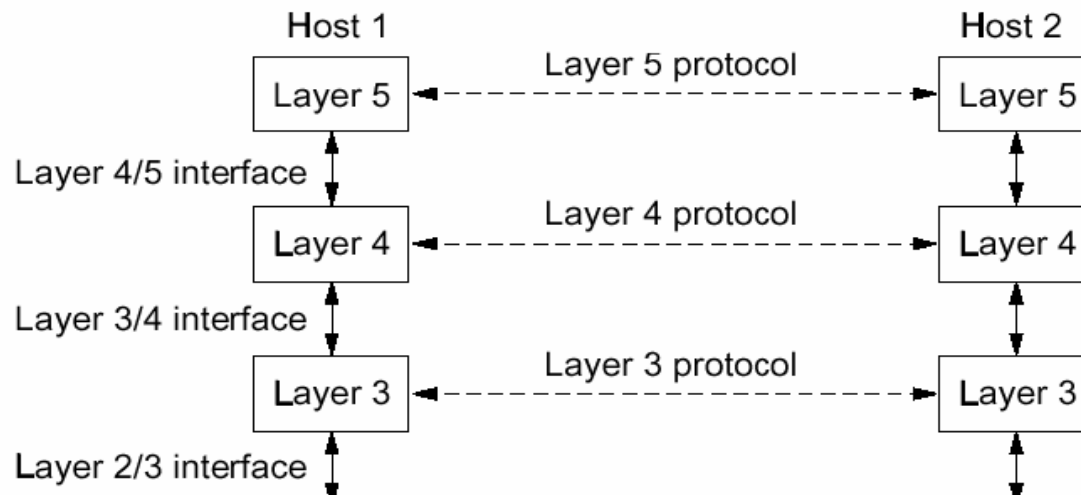
# Protocollo

- ▶ **Sintassi:** insieme e struttura dei comandi e delle risposte, formato dei messaggi
- ▶ **Semantica:** significato dei comandi, delle azioni, delle risposte da effettuare al momento della trasmissione e ricezione dei messaggi
- ▶ **Temporizzazione:** specifica delle possibili sequenze temporali di emissione dei comandi e dei messaggi, nonché delle eventuali risposte



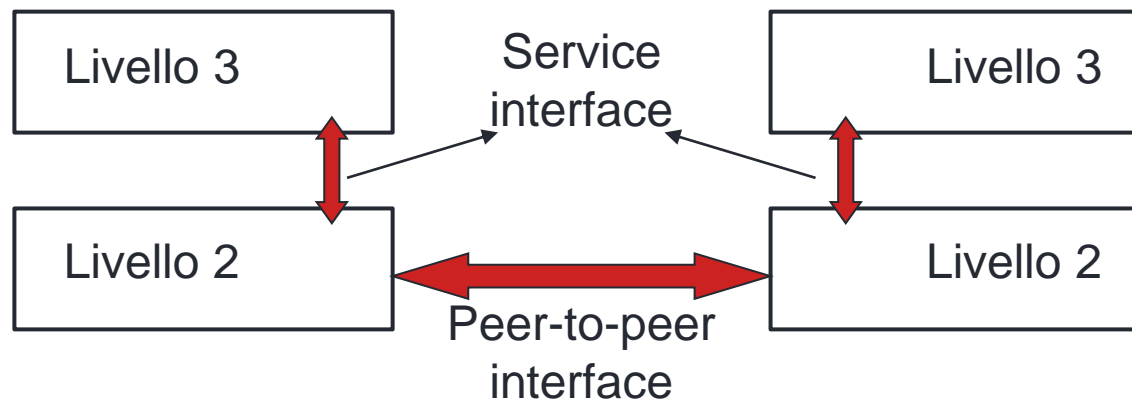
# Protocollo

- Ciascun protocollo ha un'interfaccia «interna» verso il livello superiore e inferiore, e un'interfaccia «esterna» verso il livello equivalente di un altro nodo



# Protocollo

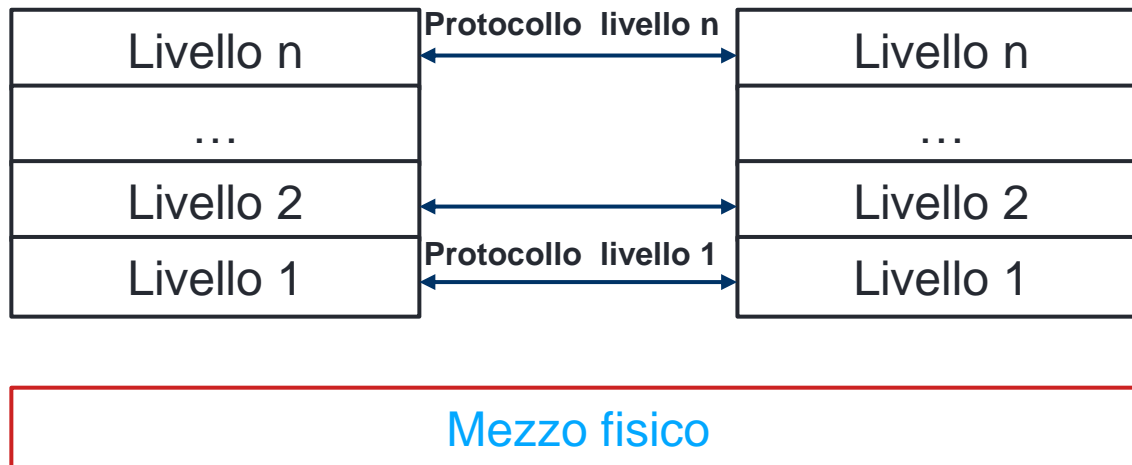
- ▶ **Service interface** (“interna”): operazioni e servizi offerti al protocollo superiore
- ▶ **Peer-to-peer interface** (“esterna”): messaggi scambiati con un livello equivalente (*peer*) sull’altro nodo





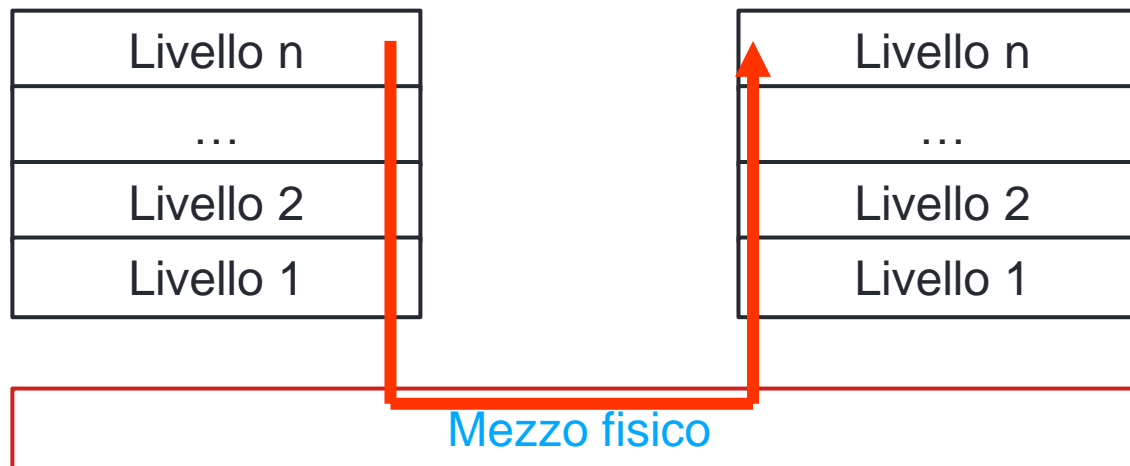
# Protocollo

- Comunicazione logica tra *peer entity* (entità allo stesso livello)



# Protocollo

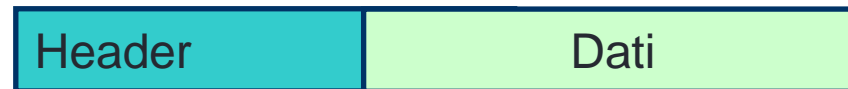
- ▶ Comunicazione fisica (indiretta)
  - ▶ Comunicazione tra *peer entity* è diretta solo a livello hardware



# Protocollo

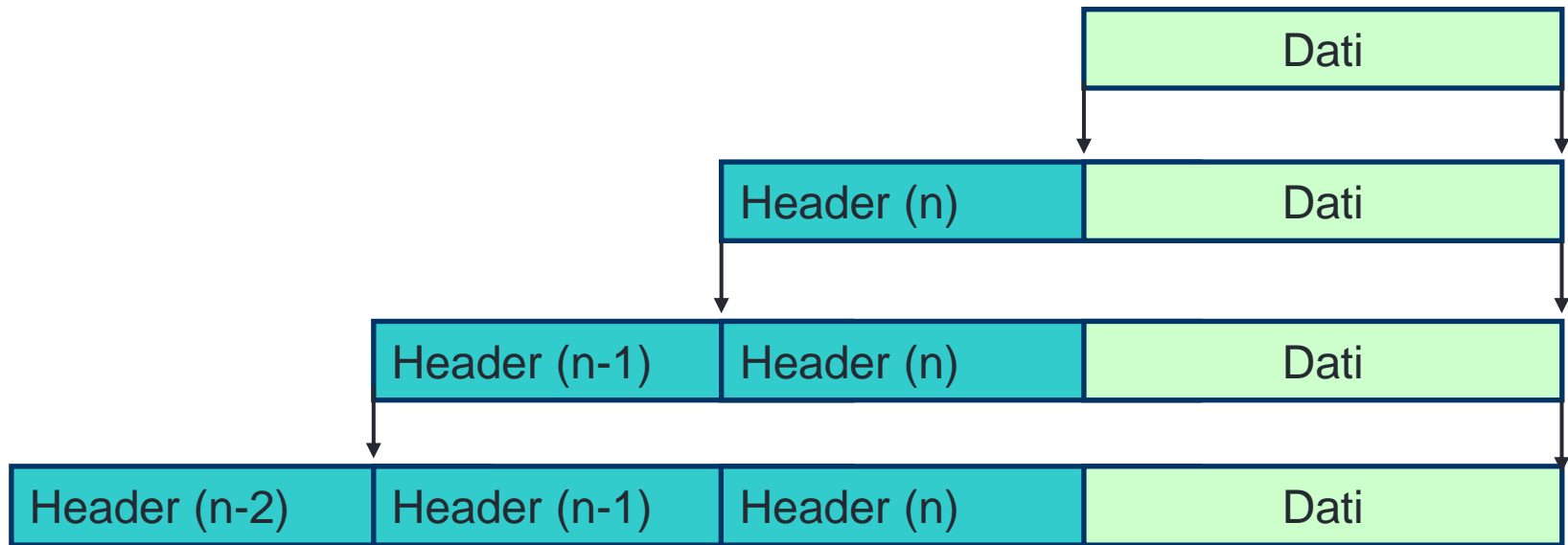
- ▶ Un messaggio si compone di
  - ▶ **Header:** Protocol Control Information (PCI)
  - ▶ **Dati:** Service Data Unit (SDU)
- ▶ L'insieme delle due è chiamata PDU (Protocol Data Unit)

## MESSAGGIO



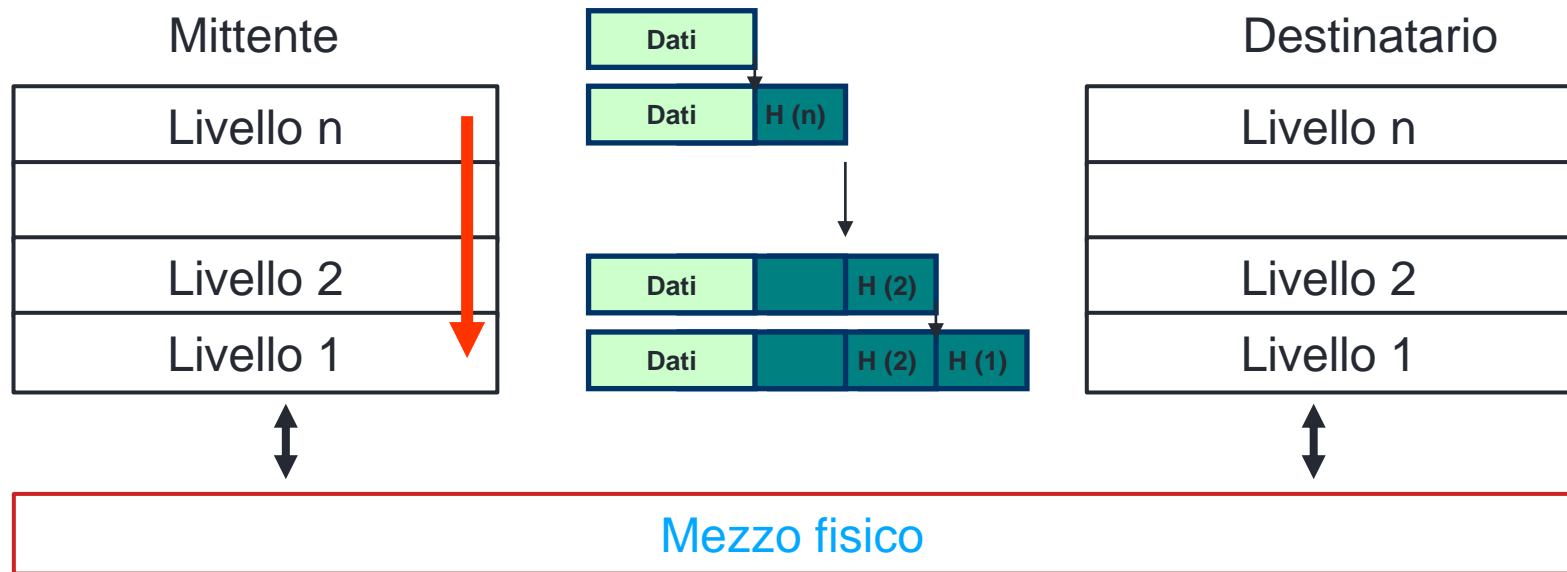
# Protocollo

## ► Incapsulamento del messaggio



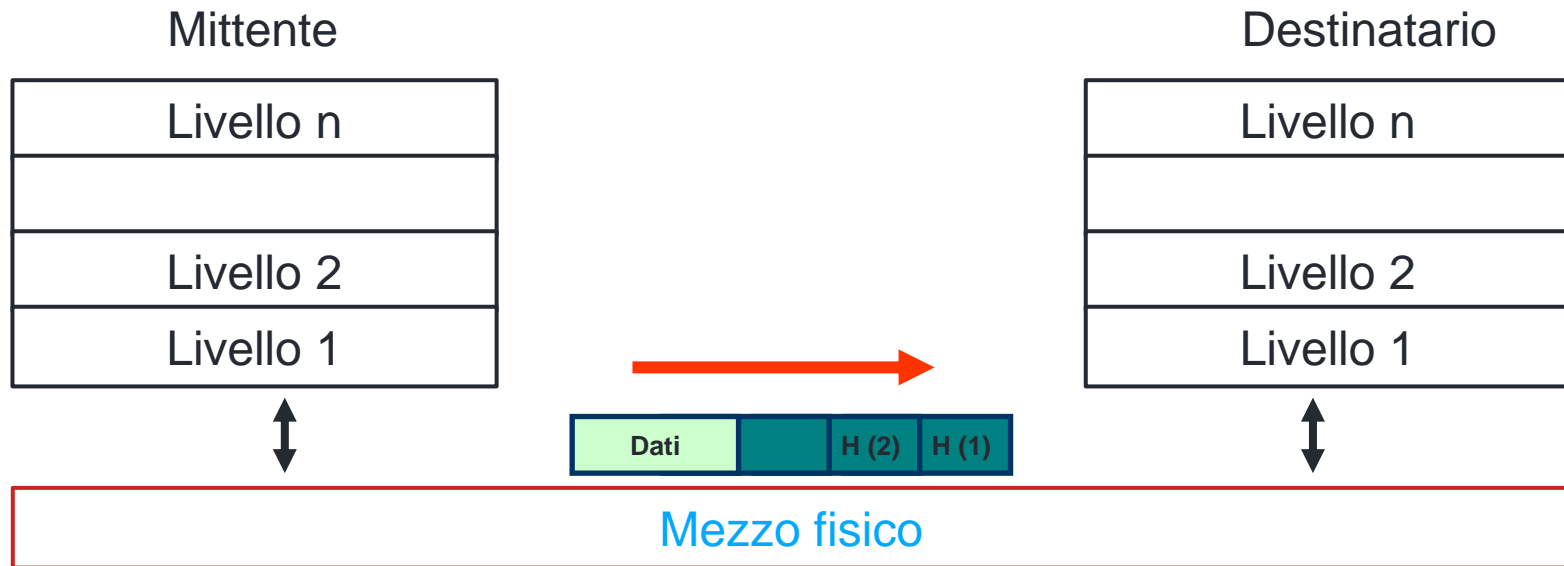
# Protocollo: Esempio comunicazione

- Costruzione e incapsulamento messaggio



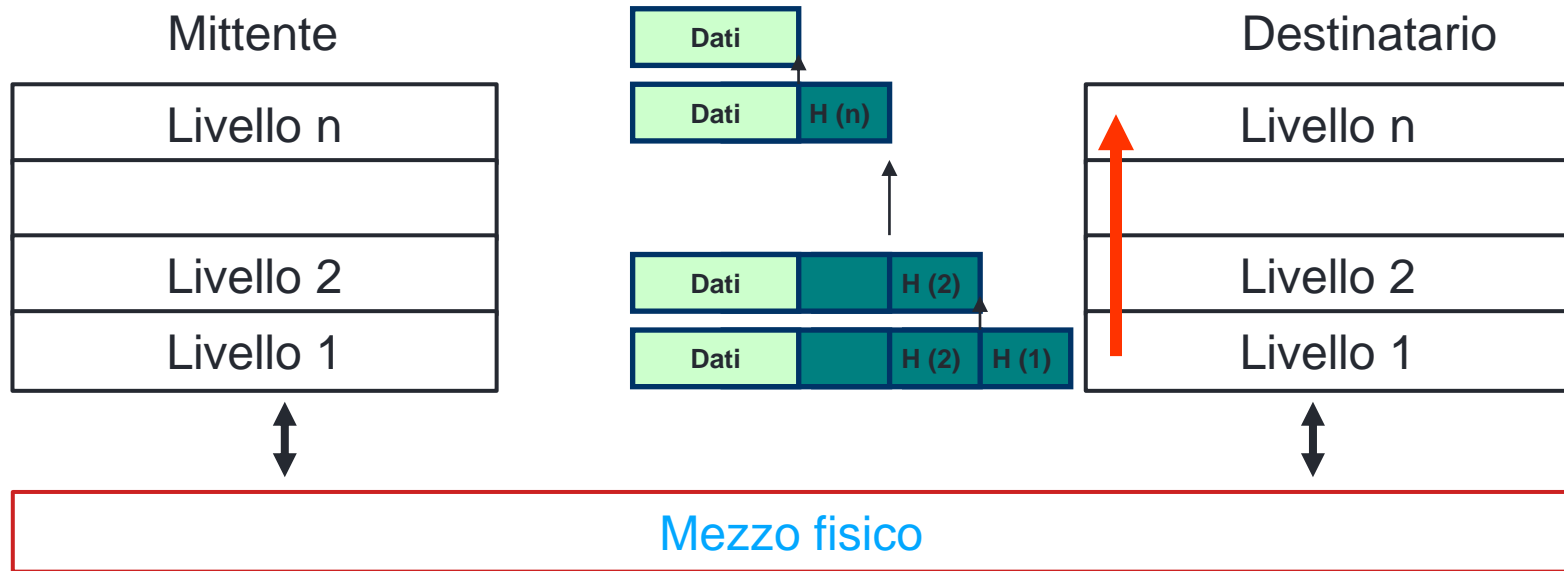
# Protocollo: Esempio comunicazione

## ► Trasferimento sul mezzo fisico



# Protocollo: Esempio comunicazione

## ► Ricezione del messaggio



# Protocollo

- ▶ La comunicazione avviene **logicamente** tra *peer*, ma in realtà attraversa tutti i livelli sottostanti, mediante **incapsulamento del messaggio** a ciascun livello
- ▶ Il sistema di comunicazione richiede un insieme di protocolli tra loro cooperanti (detti **protocol suite** o **protocol stack**)



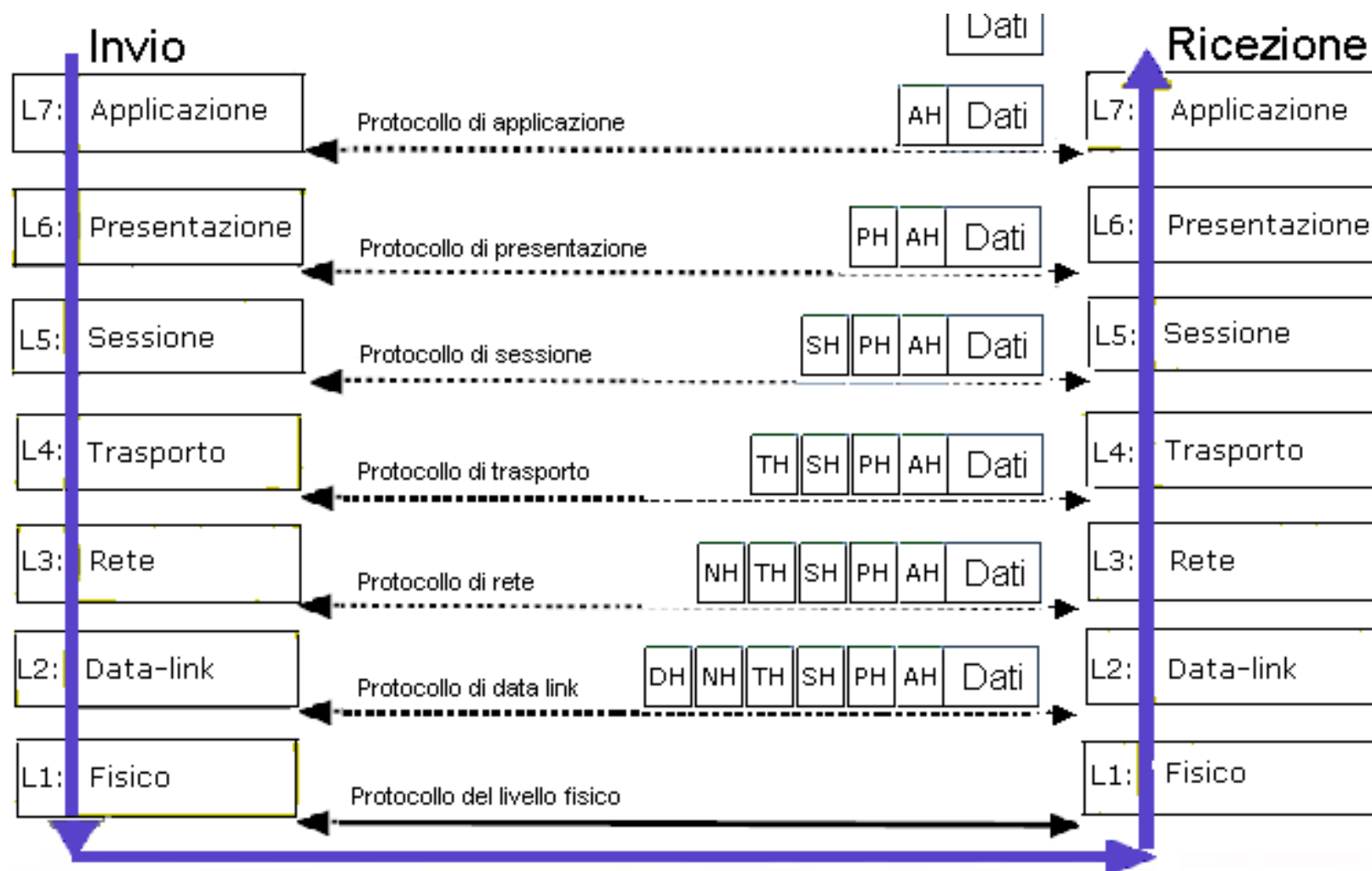


# Standard ISO-OSI

- ▶ E' stato formulato come modello per la struttura di un sistema di comunicazione
- ▶ All'interno possono essere definiti standard specifici per ciascuno strato
  - ▶ Non è detto che ci debba essere uno specifico standard per ciascuno strato
  - ▶ Esistono una serie di standard associati ad ogni strato ognuno dei quali offre svariate funzionalità

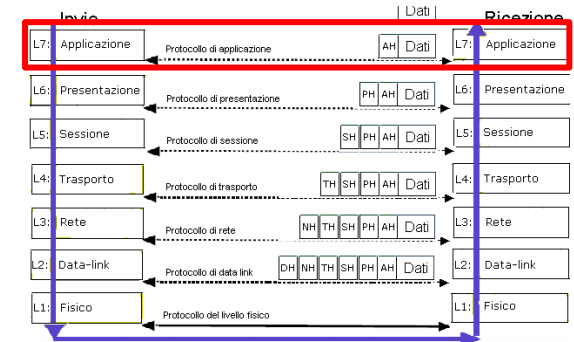


# Standard ISO-OSI



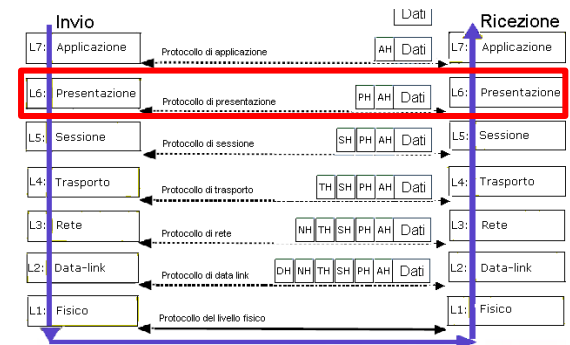
# Standard ISO-OSI: Applicazione

- ▶ Fornisce all'interfaccia verso l'utente una gamma di servizi distribuiti sulla rete
- ▶ L'accesso ai servizi avviene tramite la chiamata di primitive simili a chiamate di sistema per dispositivi locali
- ▶ Il funzionamento del sottosistema di comunicazione è del tutto trasparente
- ▶ Protocolli: HTTP, SMTP, FTP, SNMP, Telnet, DNS



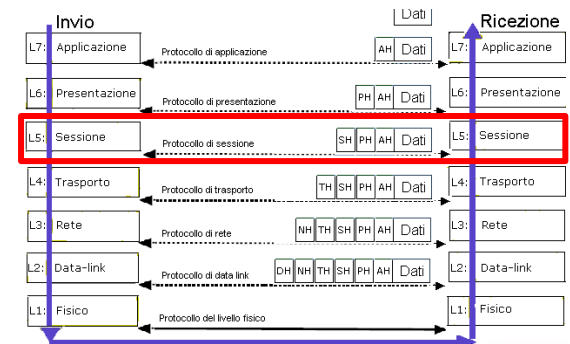
# Standard ISO-OSI: Presentazione

- ▶ Riguarda la rappresentazione dei dati durante il trasferimento (sintassi)
- ▶ Negozia e seleziona la rappresentazione di trasferimento appropriata eseguendo la necessaria conversione
- ▶ Occorre quindi una rappresentazione comprensibile ad entrambe le parti
- ▶ Spesso contiene anche funzioni sulla sicurezza dei dati (ad es., crittografia)
- ▶ Protocolli: MIME, SSL, XDR



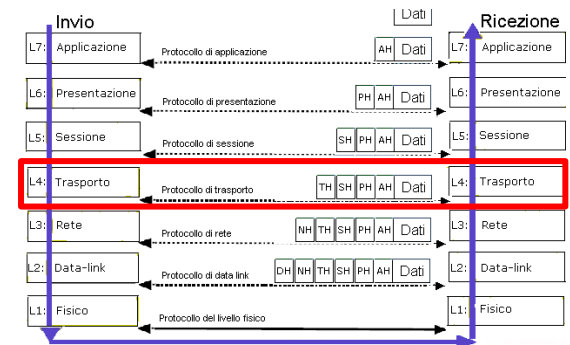
# Standard ISO-OSI: Sessione

- ▶ Consente a due entità di organizzare e sincronizzare il loro dialogo
- ▶ E' responsabile della creazione e della terminazione di un canale di comunicazione
- ▶ Gestisce l'interazione sia full-duplex (doppio senso simultaneo) che half-duplex (doppio senso alternato)
- ▶ SOCKS, NetBIOS, Session establishment in TCP



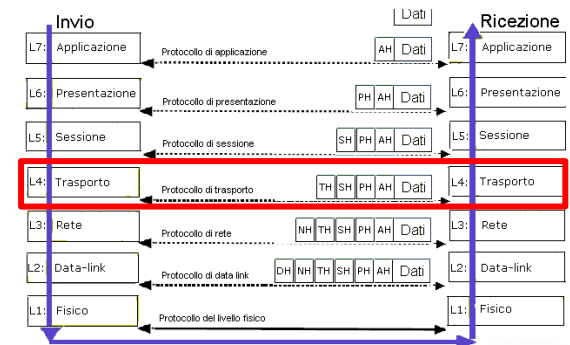
# Standard ISO-OSI: Trasporto

- ▶ Funge da interfaccia tra gli stati superiori e gli altri
- ▶ Fornisce un sistema di trasferimento dei messaggi indipendente dal tipo di rete sottostante
- ▶ Controllo dell'integrità (trasmissione affidabile)
- ▶ Riordinamento dei pacchetti rimescolati



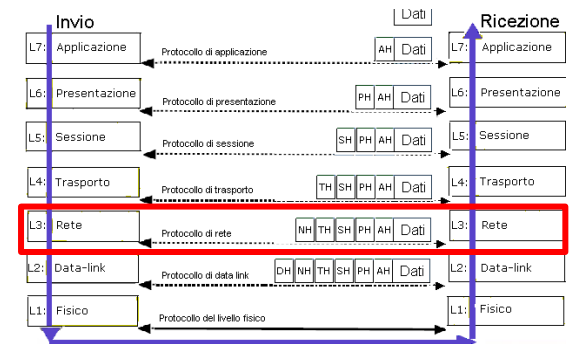
# Standard ISO-OSI: Trasporto

- ▶ Trasporto
  - ▶ senza errori
  - ▶ in sequenza
  - ▶ nessuna perdita
  - ▶ nessun duplicato
  - ▶ qualità del servizio
- ▶ Protocolli: TCP, UDP



# Standard ISO-OSI: Rete

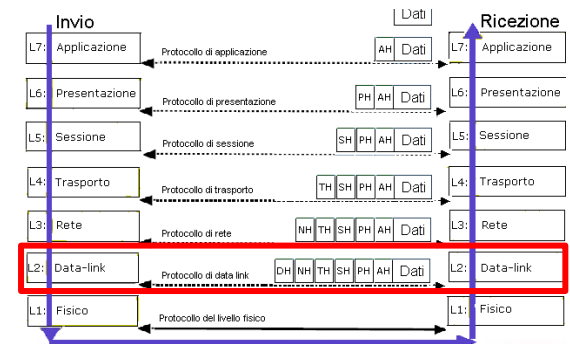
- ▶ È responsabile dell'apertura e della chiusura di una connessione
- ▶ Offre funzionalità quali: inoltra sulla rete, indirizzamento e controllo di flusso (in alcuni casi)
- ▶ In caso di inter-reti questo strato si occupa di armonizzare le diverse reti (internetworking)
- ▶ Protocolli: IP, ICMP, IPsec, ARP, RIP, OSPF





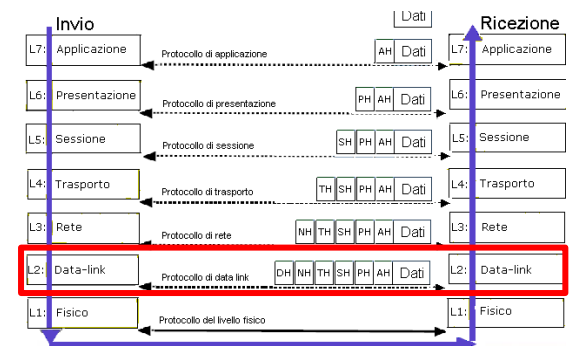
# Standard ISO-OSI: Data link

- ▶ Si basa sulla connessione fisica fornita dal tipo di rete per garantire un sistema affidabile di trasferimento delle informazioni
  - ▶ Raggruppa i bit ricevuti dallo strato superiore e li spedisce in frame
  - ▶ Responsabile della individuazione degli errori e dell'eventuale ritrasmissione, e del controllo del flusso
- ▶ Permette di connettersi e recapitare dati a un nodo adiacente (connesso da un cavo, fibra...) nella rete



# Standard ISO-OSI: Data link

- ▶ Normalmente vengono forniti due tipi di servizio:
  - ▶ **Senza connessione:** ogni frame o porzione di messaggio è una entità a se stante. Si utilizza l'approccio **best-try** se vengono individuati degli errori in un frame viene scartato e basta
  - ▶ **Orientato alla connessione:** cerca di fornire un trasferimento delle informazioni esente da errori



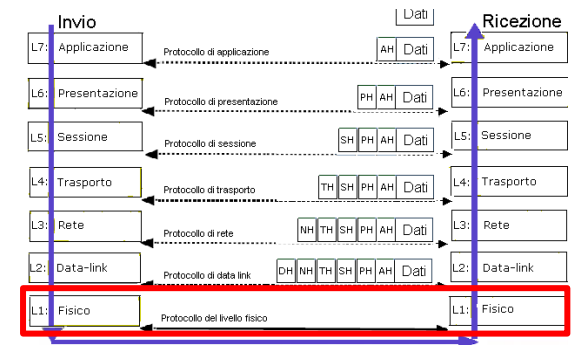
# Standard ISO-OSI: Data link

- ▶ Data link (ad es., Ethernet)
  - ▶ Trasforma una semplice funzionalità di trasmissione in una linea senza errore di trasmissione
  - ▶ Con reti di tipo broadcast serve a controllare l'accesso al canale condiviso
- ▶ Ha due responsabilità principali (sottolivelli)
  - ▶ Controllo di accesso al mezzo (MAC)
  - ▶ Rilevazione e correzione degli errori, gestione e indirizzamento (di conseguenza i livelli più alti possono presupporre una trasmissione senza errori) (LLC)
- ▶ Protocolli: PPP, SLIP



# Standard ISO-OSI: Fisico

- ▶ Riguarda le interfacce fisiche tra il calcolatore e la rete
- ▶ Fornisce i mezzi per trasmettere un flusso seriale di bit tra i due dispositivi comunicanti



# OSI semplificato

- ▶ Per renderlo più aderente alla situazione reale all'implementazione di protocolli presenti oggi sulla rete
- ▶ In sostanza OSI alla fine degli anni 80 ha subito una tendenza alla riduzione dei livelli per l'influenza di Internet



# Riduzione delle interfacce

- ▶ Il modello ISO/OSI prevede 7 livelli, con 6 interfacce tra un livello e l'altro
- ▶ Ogni interfaccia si occupa di una serie di compiti ben precisi
- ▶ Elevata modularità ma
  - ▶ Non corrisponde all'evoluzione delle applicazioni di rete
  - ▶ Tendenza alla riduzione nel numero delle interfacce dovuta all'influenza dei protocolli di Internet (TCP/IP)



# OSI vs TCP/IP model

ISO OSI

TCP/IP

Application	Application
Presentation	
Session	
Transport	Trasport
Network	Network
Datalink	Datalink
Physical	Physical



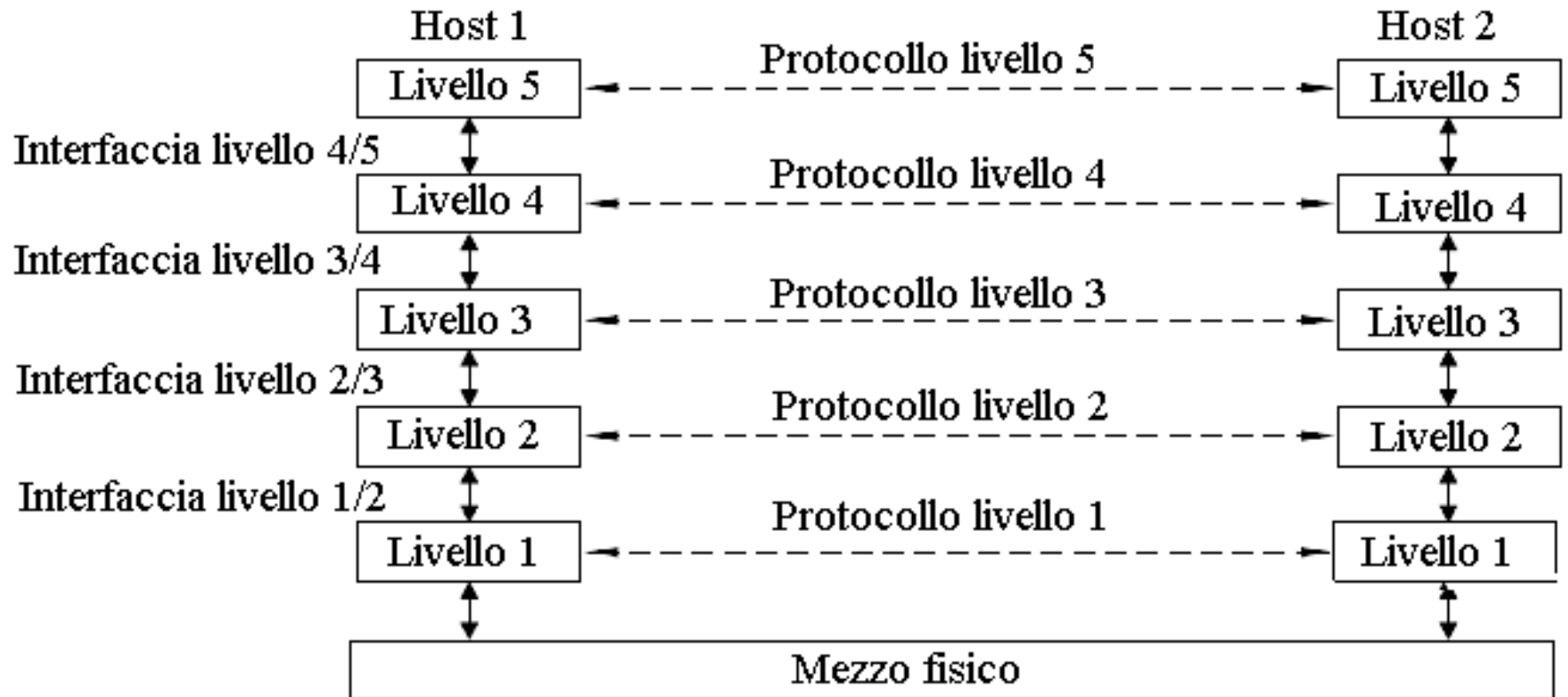
# OSI semplificato (TCP/IP model)

- ▶ Livello applicazione: applicazioni vere e proprie
- ▶ Livello trasporto: aggiunge funzionalità quali affidabilità e tolleranza ai guasti
- ▶ Livello rete: comunicazione di base tra reti diverse

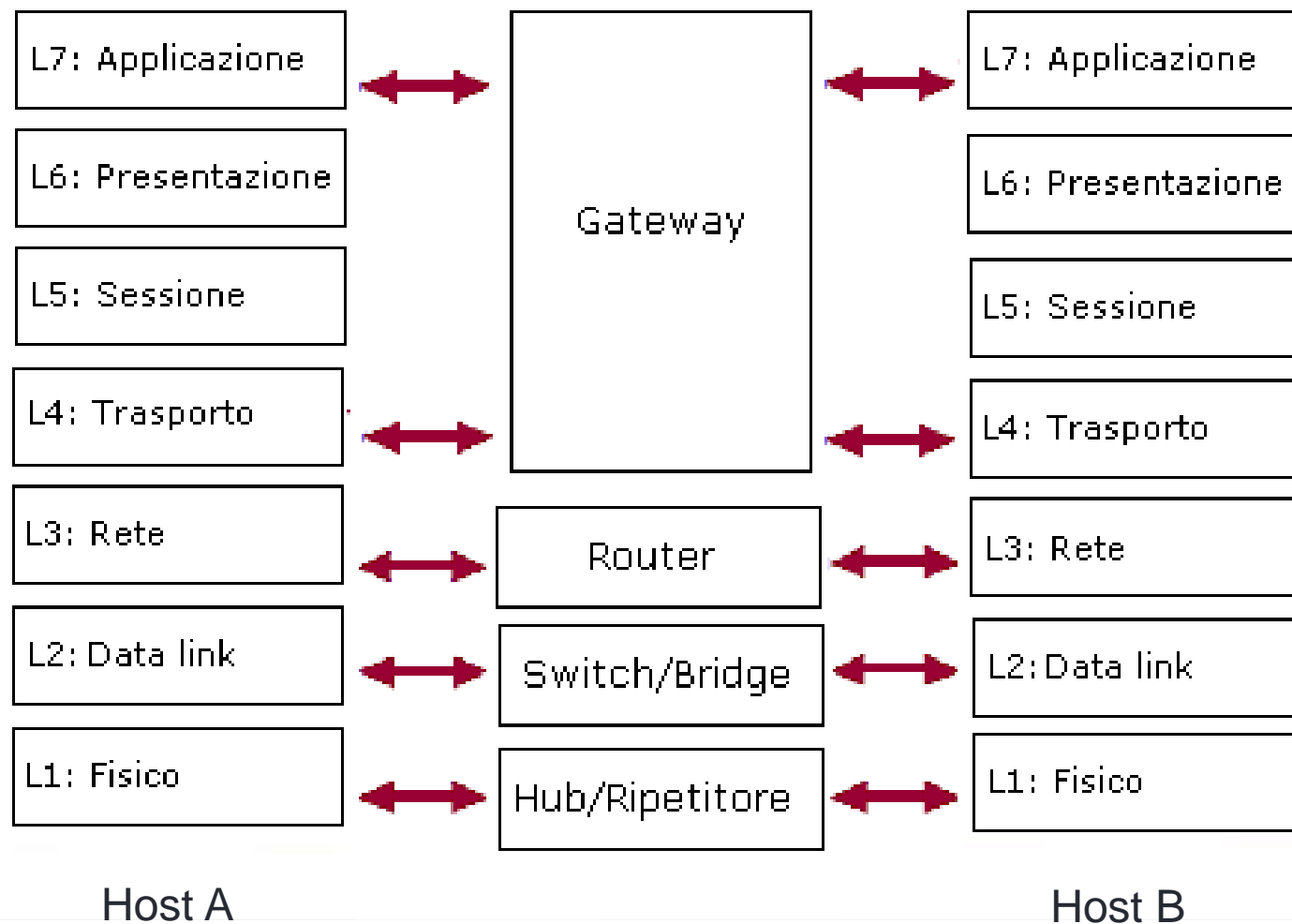




# Divisione in livelli di responsabilità



# Dispositivi d'interconnessione e ISO/OSI



# Dispositivi d'interconnessione

- ▶ **Ripetitore:** dispositivo a livello fisico che ripristina i dati e i segnali di collisione
  - ▶ Un amplificatore digitale
- ▶ **Hub:** ripetitore multi-porta a livello fisico, con rilevazione di guasti
- ▶ **Bridge:** dispositivo a livello di data link che connette due o più domini di collisione
- ▶ **Switch:** bridge multiporta con cammini attivi in parallelo



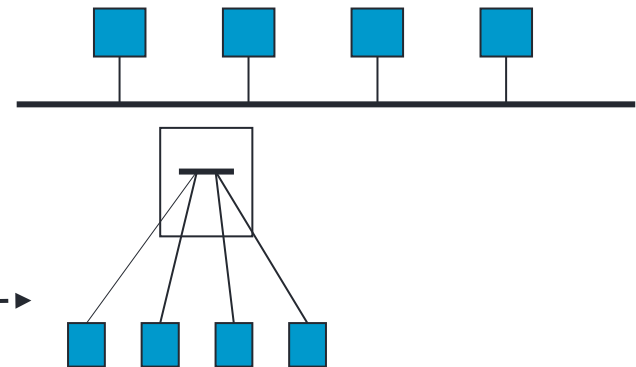
# Ripetitore

- ▶ Dispositivo di basso livello che amplifica o rigenera segnali deboli
- ▶ Viene usato per aumentare la lunghezza totale della rete
- ▶ Agisce a livello fisico e consente a tutto il traffico di attraversare i segmenti LAN



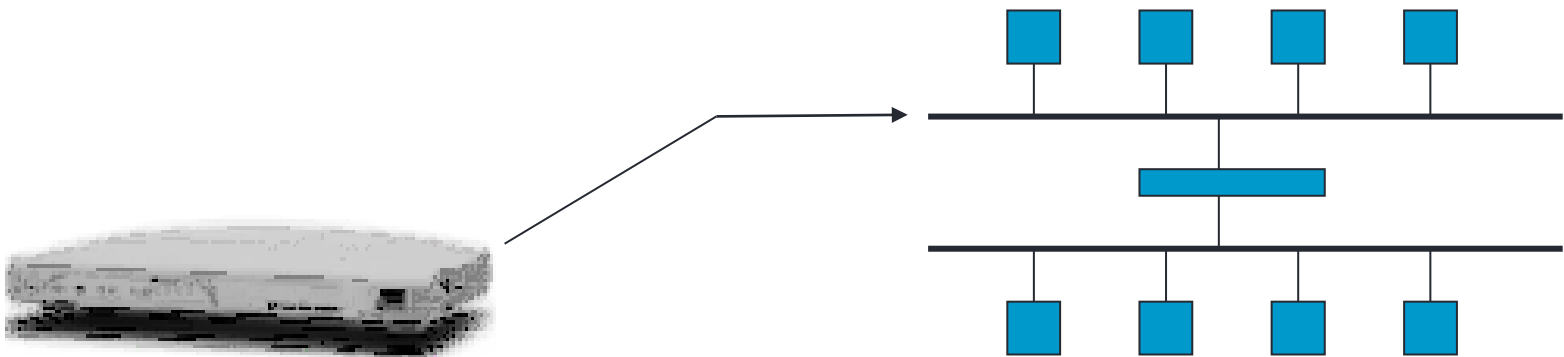
# Hub

- ▶ Punto di convergenza in cui i dati arrivano da una o più direzioni e vengono inoltrati in una o più direzioni
  - ▶ Gli hub sono anche ripetitori, con funzionalità di rilevazione dei guasti
- ▶ Connettono dispositivi o stazioni formando una LAN Ethernet
  - ▶ bus logico
  - ▶ stella fisica



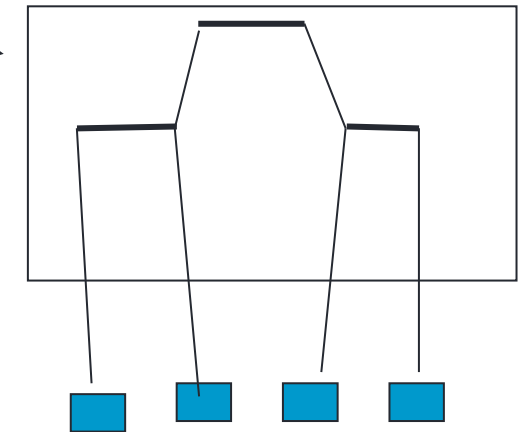
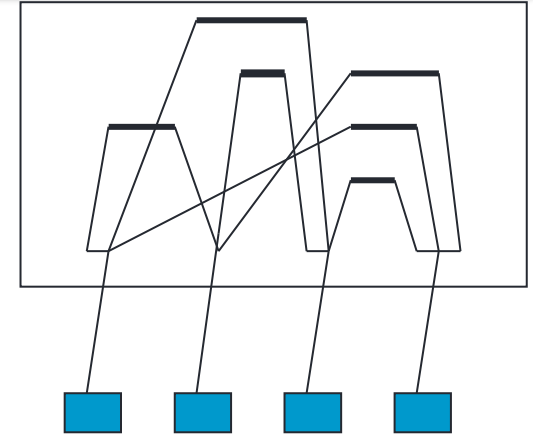
# Bridge

- ▶ Connette due LAN che usano lo stesso protocollo (ad esempio, Ethernet o Token Ring)
- ▶ Funziona al livello di data link, copiando i data frame da una rete a quella successiva lungo il cammino di comunicazione
- ▶ Può apportare cambiamenti minori al frame prima di inoltrarlo (ad esempio, aggiungendo e cancellando alcuni campi dall'intestazione del frame)



# Switch

- ▶ Tipo speciale di hub con intelligenza
  - ▶ Legge il MAC address del PC destinatario e usa questa informazione per inoltrare il frame
- ▶ Un dispositivo di rete che elabora i pacchetti ai livelli 2 e 3
- ▶ Evita collisioni attraverso cammini indipendenti
- ▶ Problemi di scalabilità per numeri elevati d'ingressi
  - ▶ **Soluzione: parallelismo parziale**
- ▶ Opera a vari livelli OSI
  - ▶ Switch di livello 2:
    - ▶ filtra e inoltra al livello di data link
    - ▶ usa gli indirizzi MAC per determinare dove vanno inviati i frame
  - ▶ Switch di livello 3:
    - ▶ inoltra i pacchetti usando le informazioni di livello 3



# Switch

- ▶ Lo switch seleziona un cammino o una linea per inviare ogni frame alla sua destinazione successiva
  - ▶ Uno switch è più semplice di un router e ha un meccanismo più veloce
- ▶ Lo switching simultaneo di pacchetti tra varie porte aumenta molto la larghezza di banda di una LAN





# Router

- ▶ Determinano il punto successivo della rete in cui deve essere inviato un pacchetto lungo il cammino verso la sua destinazione finale
- ▶ Usano le informazioni del protocollo del livello 3 di rete presenti all'interno di ogni pacchetto per instradarlo da una LAN all'altra
  - ▶ Ciò significa che un router deve saper riconoscere tutti i protocolli del livello di rete che possono essere usati sulle reti che collega
  - ▶ Soprattutto usati in rete TCP/IP: utilizzano indirizzi IP per l'instradamento
- ▶ Comunicano tra loro per determinare il cammino migliore attraverso più LAN per aumentare la velocità e ridurre il traffico di rete



# Gateway

- ▶ Usato per connettere reti che usano protocolli diversi
- ▶ Operano a livello di rete e superiori dell'ISO/OSI
- ▶ Per comunicare a un host su un'altra rete bisogna configurare un router verso quella rete
- ▶ Se non esiste si usa un gateway (default IP router)
- ▶ Se non esiste il gateway configurato allora solo comunicazioni in rete locale
- ▶ Gateway riceve dati da una rete con uno stack di protocollo, rimuove questo stack di protocollo e ricostruisce il messaggio usando il protocollo della rete di destinazione
  - ▶ Gateway tra un Systems Network Architecture (SNA) host e computer sulla rete TCP/IP
  - ▶ A packet assembler/disassembler (PAD) che fornisce connettività tra una LAN e una X.25 packet-switching network



# Conclusioni

- ▶ Reti di calcolatori: LAN, MAN, WAN, Internet
- ▶ Internet punto di riferimento per le reti di calcolatori
- ▶ ISO/OSI model, modello concettuale che rappresenta il funzionamento interno di un sistema di comunicazione
- ▶ TCP/IP model, mapping del modello ISO/OSI sulla rete Internet
- ▶ Dispositivi di interconnessione



# Livello di rete: IP

- ▶ Protocollo per la consegna dei pacchetti da un host mittente a un host destinatario
  - ▶ Identificativo univoco di ciascun host (indirizzo IP)
  - ▶ Comunicazione logica tra host
- ▶ Ma
  - ▶ *Privo di connessione*: ogni pacchetto è trattato in modo indipendente da tutti gli altri
  - ▶ *Non affidabile*: la consegna non è garantita (i pacchetti possono essere persi, duplicati, ritardati, o consegnati senza l'ordine di invio)
  - ▶ *Consegna con impegno*: tentativo di consegnare ogni pacchetto (l'inaffidabilità deriva dalle possibili congestioni della rete o guasti dei nodi/router)



# Protocollo IP

- Il protocollo IP fornisce un servizio datagram connectionless e inaffidabile
- Il termine inaffidabile significa che non ci sono garanzie che un pacchetto IP giunga a destinazione (servizio best effort)
- Il termine connectionless significa che il protocollo IP non mantiene alcuna informazione di stato circa i pacchetti inoltrati
- Ciascun pacchetto è trattato indipendentemente da tutti gli altri
  - I datagrammi IP possono essere consegnati fuori sequenza



# Header IP

4bit version	4b head. length	8bit type of service (TOS)	16bit total length (in bytes)	
16bit identification			3bit flags	13bit fragment offset
8bit time to live (TTL)		8bit protocol	16bit header checksum	
32bit source IP address				
32bit destination IP address				
options, data				

# Header IP

- ▶ Versione (IPv4/IPv6)
- ▶ Lunghezza IP Header (numero di parole a 32-bit nello header, di solito 5)
- ▶ Type of Service (di solito 0, può contenere richieste di Quality of Service al livello 2)
- ▶ Dimensione Datagram (lunghezza in byte, header+dato)
- ▶ Identificazione (numero a 16 bit che insieme all'indirizzo mittente identifica univocamente il pacchetto nel caso debba essere frammentato e ricomposto)



# Header IP

- ▶ Fragmentation Offset (se è di un frammento, spiazzamento in byte dall'inizio del pacchetto originale. Può essere impostato dai router che eseguono la frammentazione)
  - ▶ Preceduto da tre bit di flag, due dei quali si riferiscono alla frammentazione
- ▶ Time To Live: numero di hop, ovvero router che il pacchetto può attraversare
  - ▶ Decrementato dai router per prevenire cicli
- ▶ Protocol Service Access Point (SAP) indica lo header che segue (e.g. 1 = ICMP; 2= IGMP; 6 = TCP; 17= UDP)
- ▶ Indirizzo IP mittente
- ▶ Indirizzo IP destinazione





# Inter-reti

- ▶ Le *inter-reti* o *reti di reti* hanno le seguenti proprietà
  1. I computer posti sulle varie reti locali che costituiscono l'inter-rete (*subnet*) possono comunicare tra loro
  2. A livello di data link le subnet possono essere eterogenee, ad esempio Ethernet su un segmento di rete e Token Ring su un altro
  3. L'inter-rete può comprendere un numero illimitato di host, nel rispetto dei limiti che stabiliscono il massimo numero di host collegabili a ciascuna subnet



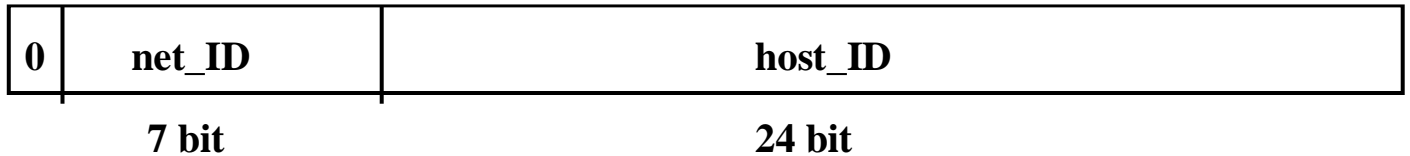
# Indirizzi IP

- ▶ Ogni interfaccia di rete ha un unico indirizzo IP di lunghezza fissa (4 byte = 32 bit)
- ▶ Indirizzi IP costituiscono una risorsa finita (per questo nascono indirizzi IPv6)
- ▶ Cinque classi di indirizzi: A, B, C, D ed E
  - ▶ D serve per trasmissioni broadcast, mentre E non è usata
- ▶ Gli indirizzi IP godono della *proprietà prefisso*
  - ▶ Il prefisso di rete (netid) di un indirizzo IP dice esattamente la rete d'appartenenza dell'interfaccia a cui è assegnato

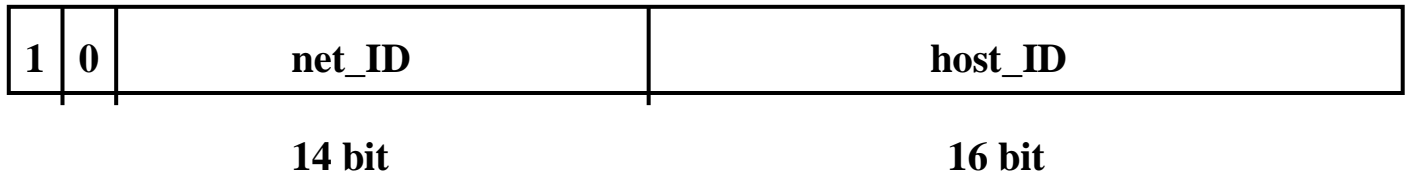


# Classi di indirizzi IP

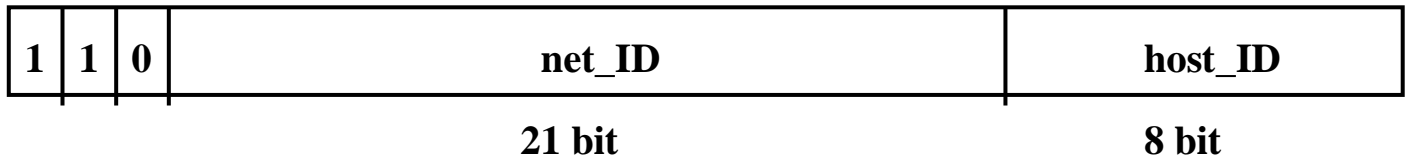
Classe A



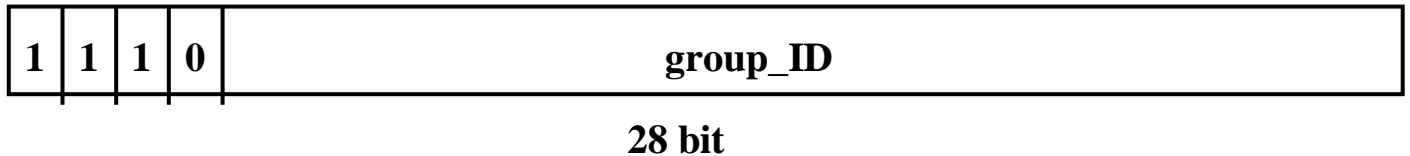
Classe B



Classe C



Classe D



Classe E



# Classi di indirizzi IP

- ▶ Le varie classi di indirizzi si distinguono (oltre che per il prefisso) per la diversa ripartizione tra l'identificativo della rete locale (net\_ID) e della scheda di rete (host\_ID)
- ▶ Ciascun tipo d'indirizzamento è attribuito ad una rete locale sulla base del numero delle macchine che la compongono
  - ▶ Indirizzo di classe C: primi 24 bit sono fissi (21 individuano la rete) e 8 sono liberi, permettendo di designare al più 256 macchine (che avranno il net\_ID in comune)
- ▶ Indirizzi impostati usando la notazione decimale puntata, che codifica un indirizzo IP con quattro numeri decimali separati da punti, (es. 196.20.44.2) ciascuno dei quali codifica il contenuto di uno dei byte dell'indirizzo
- ▶ Il valore decimale del primo byte può essere usato per identificare la classe
  - ▶ Fino a 127 indica la classe A, da 128 a 191 una classe B, da 192 a 219 una classe C e così via



# Maschere di sottorete

- ▶ La divisione in classi degli indirizzi IP fornisce tre modelli standard di suddivisione dell'indirizzo IP a 32 bit tra l'identificativo della rete e quelli degli host
- ▶ Non sempre però questa ripartizione è la più pratica
- ▶ Si può ottenere una diversa ripartizione tra `host_id` e `net_id` corredando ogni indirizzo IP di una maschera di sottorete (*subnet mask*), definita come il *numero binario che messo in AND con l'indirizzo IP dà il prefisso effettivo*



# Maschere di sottorete

- ▶ Alle tre classi di indirizzi IP sono associate maschere di default

Classe	Maschera di default
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

- ▶ Applicando le maschere di default si ottiene un prefisso equivalente al net\_id di classe



# TCP/IP

- ▶ Il Transmission Control Protocol/Internet Protocol (TCP/IP) è un software di rete disponibile che permette alle applicazioni di comunicare tramite un protocollo instradabile, lo stesso usato su Internet
- ▶ TCP/IP definisce una porzione di header fisso e tre header speciali
  - ▶ Uno semplice per il best effort (User-Datagram, UDP)
  - ▶ Uno complesso per il servizio reliable flow (TCP)
  - ▶ Uno per messaggi di controllo (Internet Control Message Protocol o ICMP)



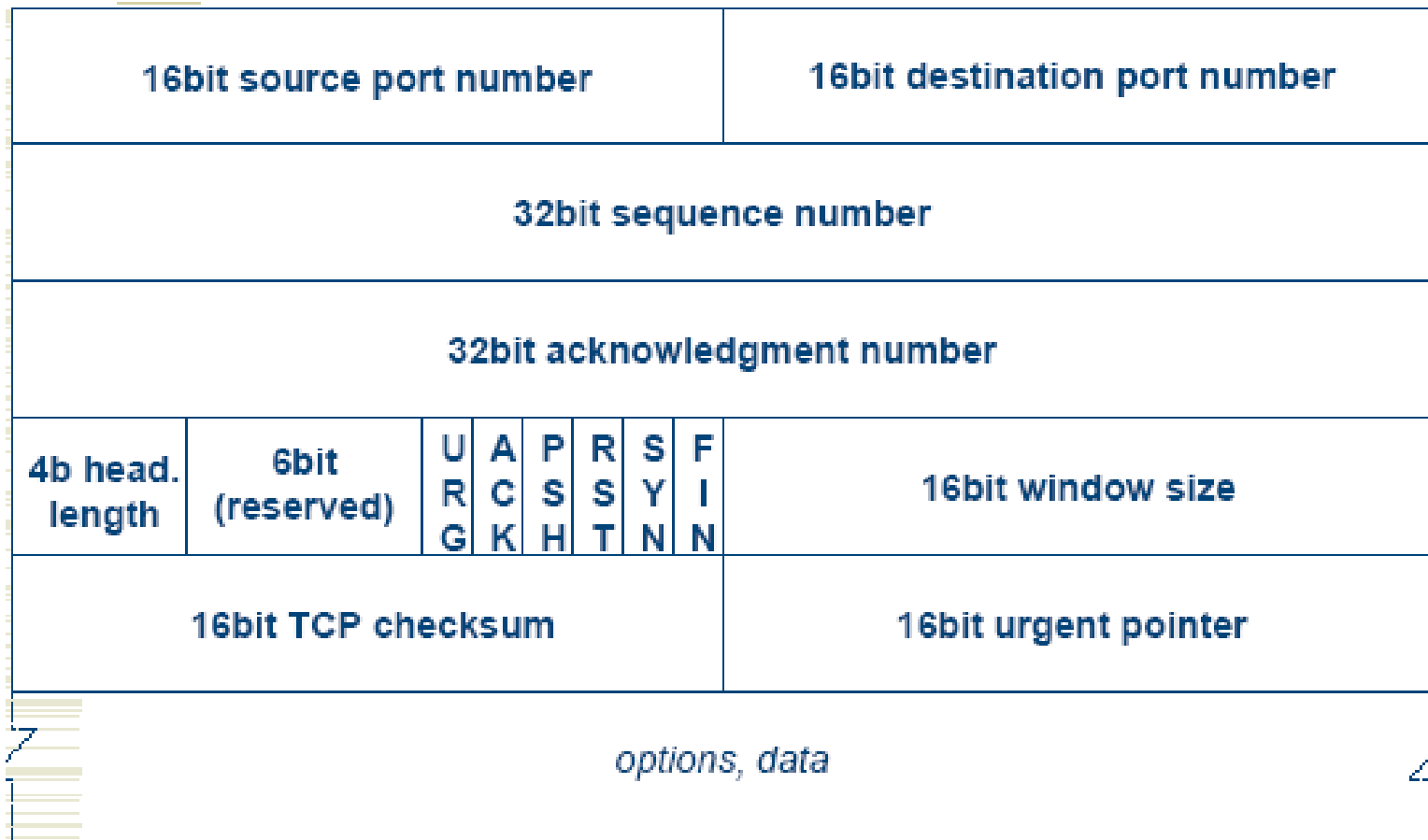
# TCP

- ▶ TCP è un protocollo orientato alla connessione
- ▶ Il software di rete che implementa TCP deve assicurare due condizioni fondamentali
  1. Il programma applicativo destinatario è attivo
  2. Tutti i pacchetti inviati dal mittente raggiungono la loro destinazione
- ▶ Per fare questo, il protocollo TCP ha bisogno di informazioni supplementari rispetto a quelle contenute nell'intestazione IP
  - ▶ Header aggiuntivo ad ogni pacchetto IP da inviare



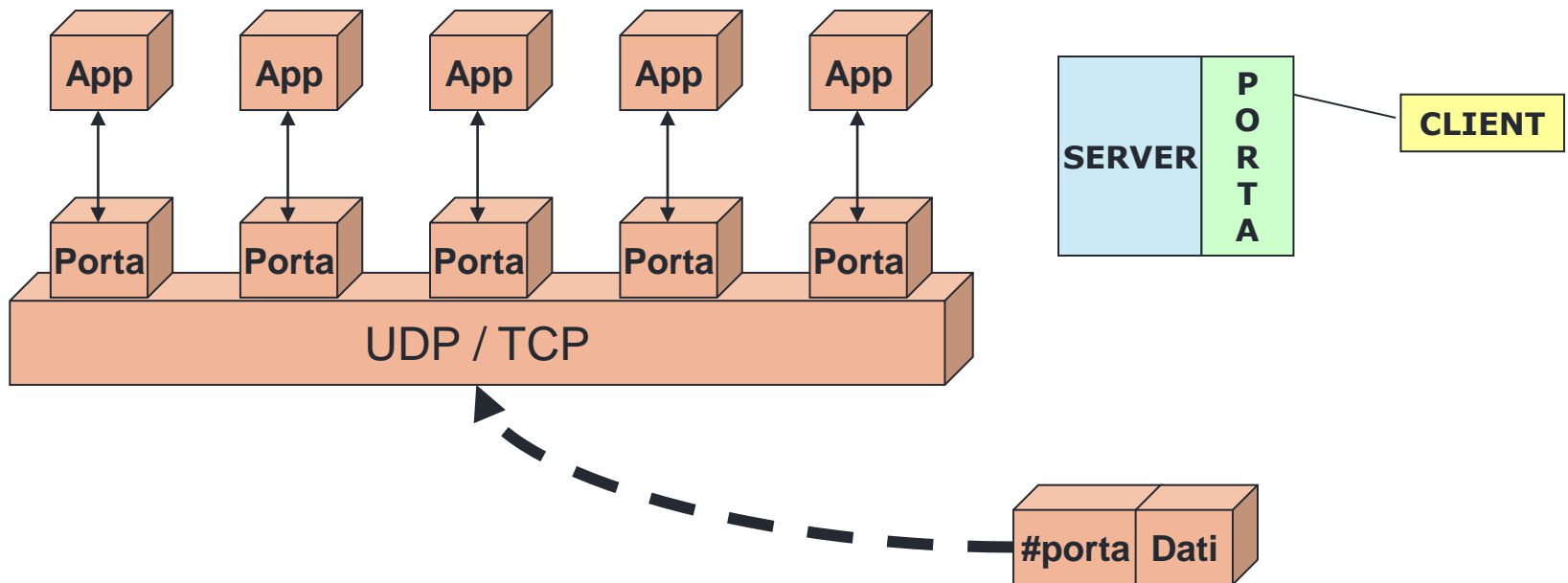


# TCP



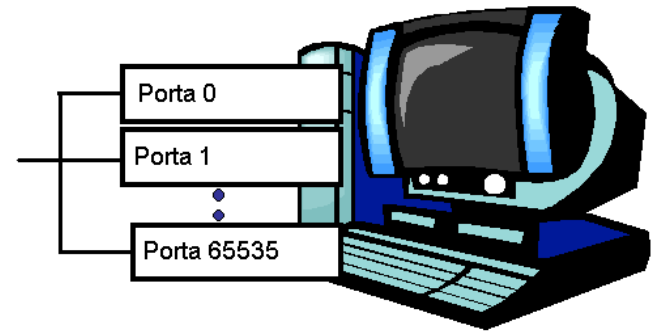
# Richiami di TCP: porte

- ▶ I protocolli TCP e UDP usano le porte per mappare i dati in ingresso con un particolare processo attivo su un computer
- ▶ Ogni socket è legato ad un numero di porta così che il livello TCP può identificare l'applicazione a cui i dati devono essere inviati



# Richiami di TCP: well-known ports

- ▶ Porte rappresentate da valori interi positivi (16 bit)
- ▶ Rappresentano punto di collegamento fra strato fisico e applicazioni; rappresentano un canale di comunicazione
- ▶ Alcune porte sono state riservate per il supporto di servizi well-known
  - ▶ ftp -> 21/tcp
  - ▶ telnet -> 23/tcp
  - ▶ smtp -> 25/tcp
  - ▶ http -> 80/tcp
  - ▶ login -> 513/tcp
- ▶ I servizi e i processi a livello utente generalmente usano un numero di porta  $\geq 1024$



# Flag TCP

- ▶ SYN: richiesta di stabilire una sessione, sempre il primo pacchetto di una comunicazione TCP
- ▶ ACK: conferma del pacchetto precedente, sia esso dati, SYN o FIN
- ▶ FIN: indica l'intenzione del mittente di terminare la sessione in maniera concordata
- ▶ RST: reset della sessione
- ▶ PSH: operazione di push, i dati vengono subito inviati al destinatario senza bufferizzarli
- ▶ URG: dati urgenti (es. CTRL+C) vengono inviati con precedenza sugli altri



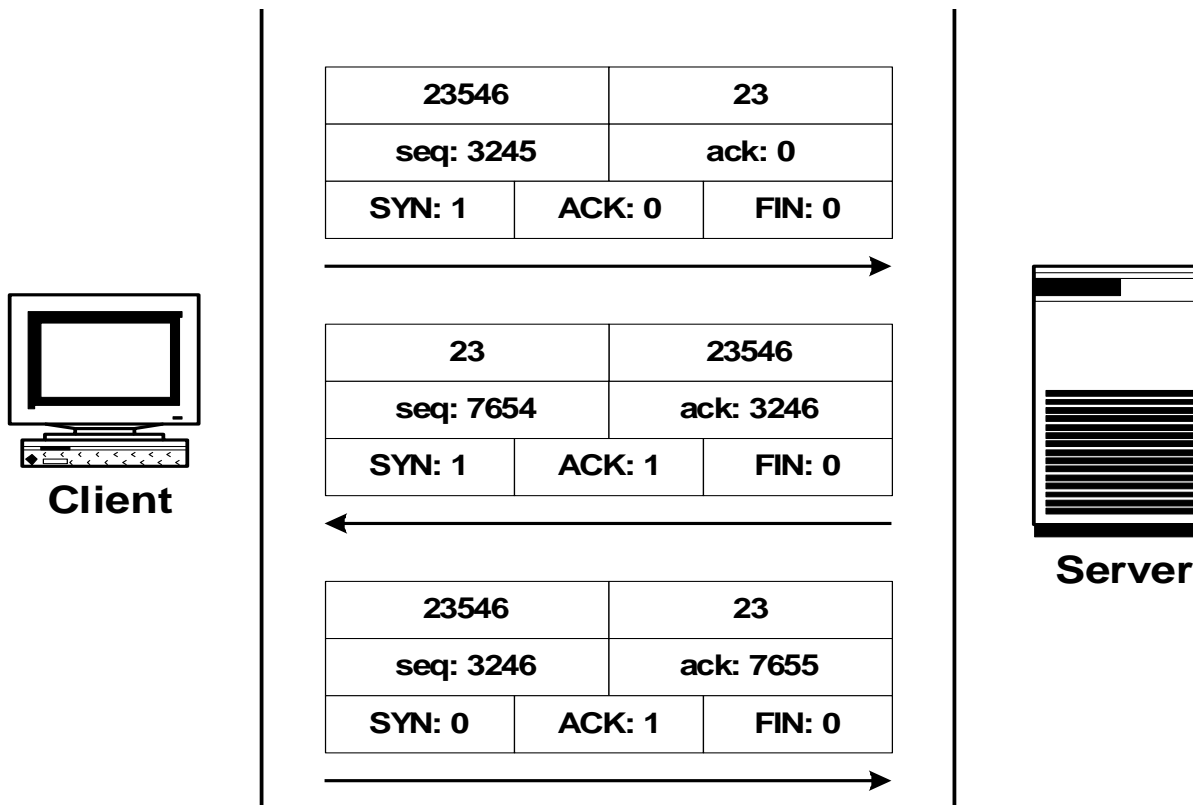
# Seq e Ack number

- ▶ Flusso di dati TCP è diviso in tanti pacchetti, che sono inviati in un ordine sequenziale numerato, all'interno di altrettanti pacchetti IP
- ▶ Il numero di sequenza nell'intestazione TCP (*sequence number*) stabilisce l'ordine che la destinazione deve usare per riassemblare i segmenti
- ▶ Quando il computer destinatario riceve un pacchetto IP risponde al mittente con un piccolo pacchetto di conferma detto ACK (ACKnowledgment) o *conferma di ricezione*
- ▶ Il numero di ciascuna conferma di ricezione coincide con il numero di sequenza del pacchetto che è stato ricevuto, più uno



# Threeway handshake

- Tecnica usata in TCP per avviare una comunicazione tra due host



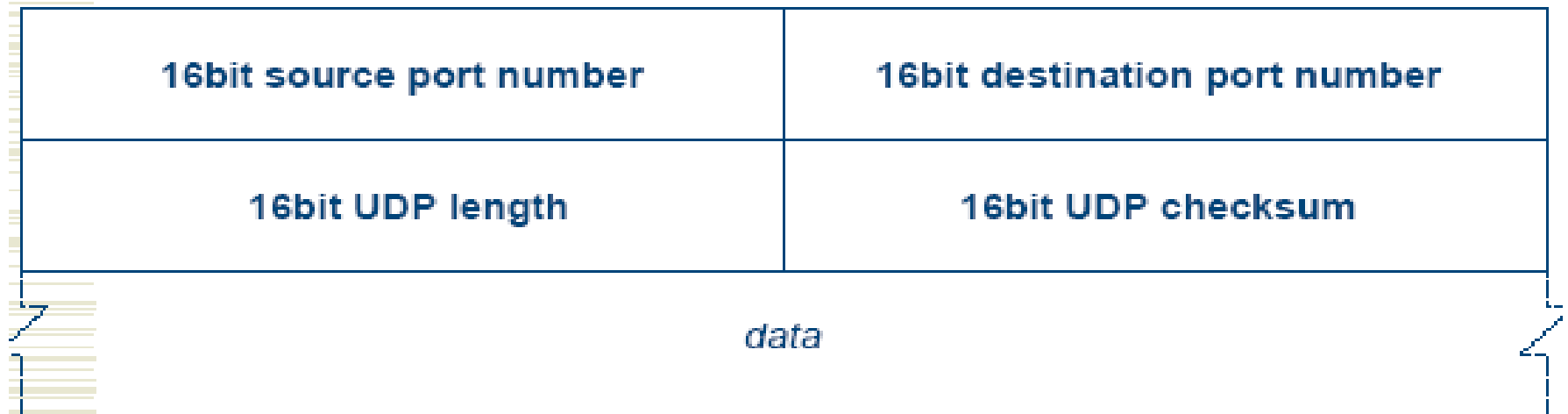
# UDP

- ▶ È un protocollo di comunicazione che lavora allo stesso livello del TCP ma non è orientato alla connessione
- ▶ Non esiste la fase di connessione come nel caso di TCP
- ▶ I pacchetti spediti senza sapere se l'applicazione del destinatario è pronta alla ricezione
- ▶ Header semplificato



# Header UDP

- ▶ UDP aggiunge al pacchetto IP solamente le porte per la comunicazione applicativa





# Confronto TCP-UDP

## ▶ TCP

- ▶ Orientato alla connessione
- ▶ Trasporto affidabile
- ▶ Controllo di flusso
- ▶ Controllo di congestione

## ▶ UDP

- ▶ Non affidabile
- ▶ Non vengono garantiti né il controllo di flusso né il controllo della congestione



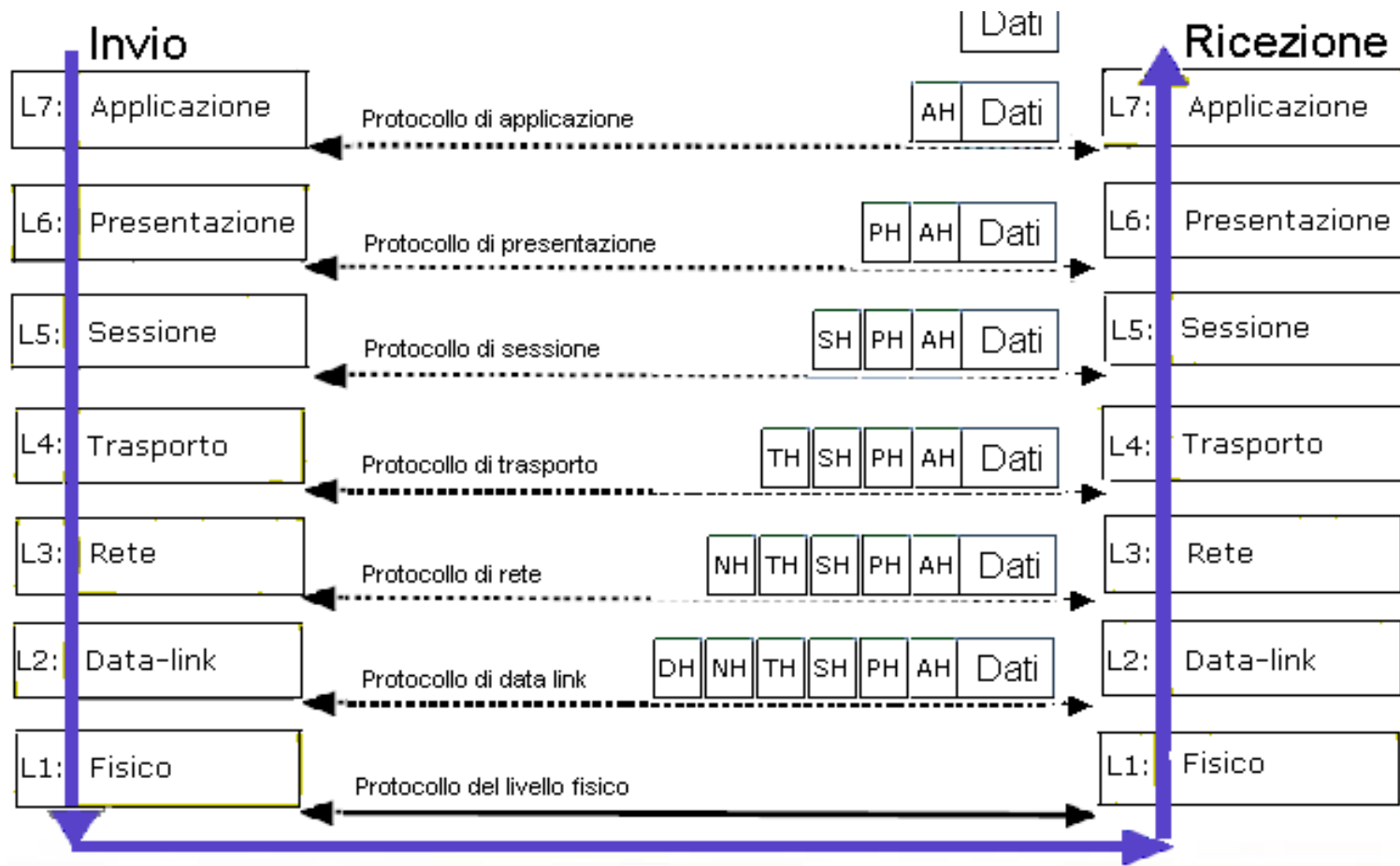
# Conclusioni

- ▶ Livello di rete
  - ▶ Protocollo IP per la consegna di pacchetti
  - ▶ Classi di indirizzi IP, indirizzi IP, maschere di sottorete
- ▶ Livello di trasporto
  - ▶ Permette alle applicazioni di comunicare tramite un protocollo instradabile
  - ▶ UDP: non orientato alla connessione
  - ▶ TCP: orientato alla connessione

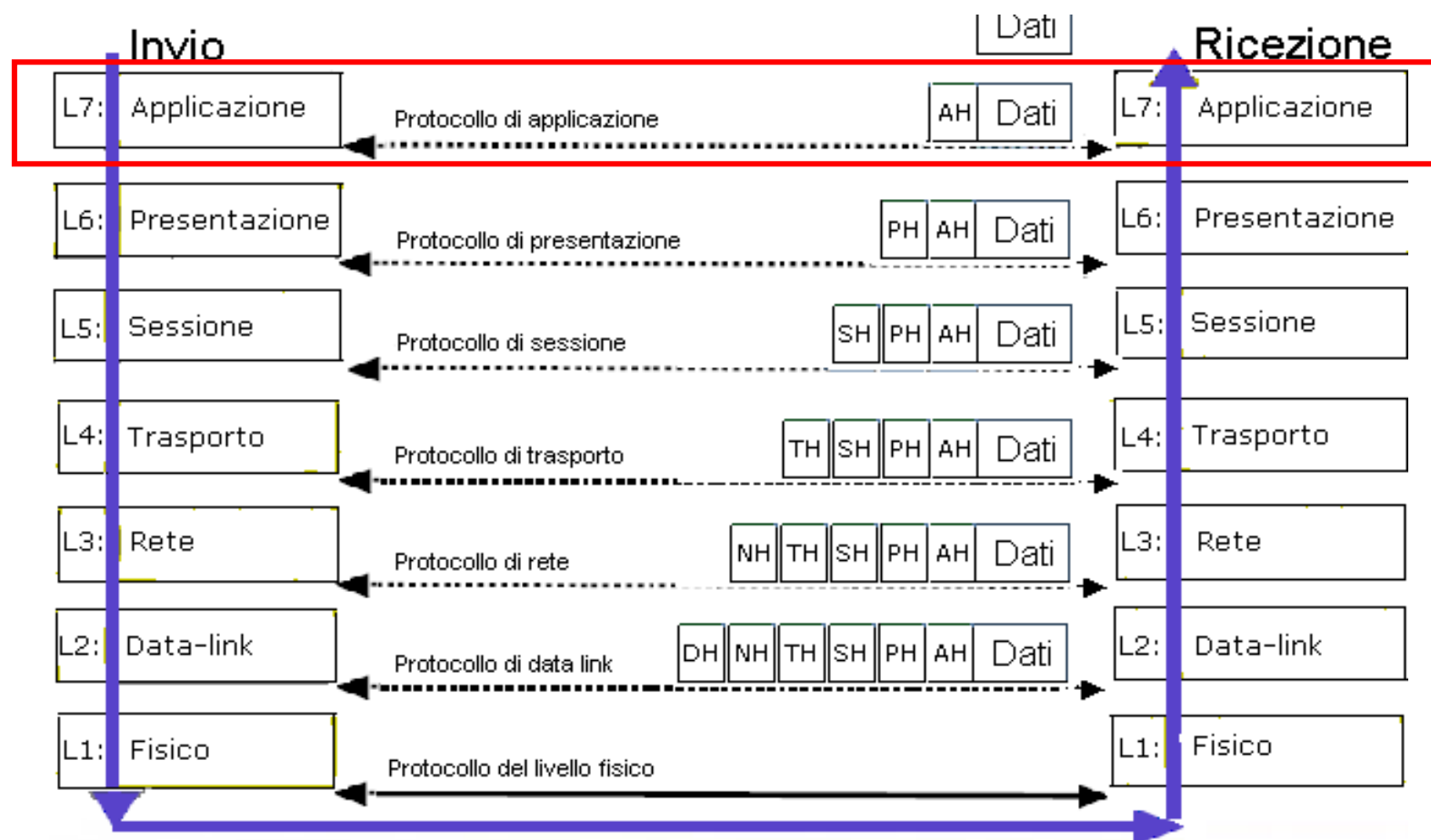




# Standard ISO/OSI



# Standard ISO/OSI



# Livello applicativo

- ▶ Il livello OSI più vicino all'utente
  - ▶ Sia il livello applicativo che l'utente interagiscono direttamente con l'applicativo software
- ▶ Interagisce con l'applicativo software che implementa il componente di comunicazione
  - ▶ Applicazioni sono *out of scope* del modello OSI
- ▶ Funzionalità del livello applicativo includono
  - ▶ Identificare i partner della comunicazione
  - ▶ Determinare la disponibilità di risorse
  - ▶ Sincronizzare la comunicazione
  - ▶ Quality of service, autenticazione, privacy



# Livello applicativo

- ▶ Identificazione dei partner di comunicazione
  - ▶ Il livello applicativo identifica e controlla la disponibilità dei partner di comunicazione per un'applicazione che deve trasmettere dati
- ▶ Disponibilità di risorse
  - ▶ Il livello applicativo decide se le risorse di rete disponibili sono sufficienti per la comunicazione
- ▶ Sincronizzazione della comunicazione
  - ▶ Le comunicazioni tra applicazioni richiedono cooperazione che è gestita dal livello applicativo



# Protocolli di livello applicativo

- ▶ Tipo dei messaggi scambiati
  - ▶ Ad es., messaggi di richiesta e risposta
- ▶ Sintassi dei tipi di messaggi
  - ▶ Quali campi e come sono distribuiti
- ▶ Semantica dei campi
  - ▶ Significato delle informazioni nei campi
- ▶ Regole che guidano i processi nell'invio di messaggi e risposte a messaggi





# Protocolli di livello applicativo

- ▶ Public-domain protocol
  - ▶ Definiti in RFC
  - ▶ Permettono interoperabilità
  - ▶ Ad es., HTTP, SMTP
- ▶ Protocolli proprietari
  - ▶ Ad es., KaZaA



# Protocolli di livello applicativo

- ▶ Esempi
  - ▶ Domain Name System (DNS)
  - ▶ Hypertext Transfer Protocol (HTTP)
  - ▶ File Transfer Protocol (FTP)
  - ▶ Simple Mail Transfer Protocol (SMTP)



# Applicazioni di rete

- ▶ E-mail
- ▶ Web
- ▶ Instant messaging
- ▶ Remote login
- ▶ P2P file sharing
- ▶ Multi-user network games
- ▶ Streaming stored video clips
- ▶ VOIP
- ▶ Real-time video conference
- ▶ Massive parallel computing



# Livello di trasporto

- ▶ Perdita di dati
  - ▶ Alcune app (ad es., audio) possono tollerare alcune perdite
  - ▶ Altre app (ad es., file transfer, telnet) richiedono un trasferimento affidabile al 100%
- ▶ Timing
  - ▶ Alcune app (ad es., VOIP, interactive game) richiedono basso delay per essere utilizzabili
- ▶ Bandwidth
  - ▶ Alcune app (ad es., multimedia) richiedono una minima quantità di banda per essere efficaci
  - ▶ Altre app (“elastic app”) usano qualunque banda ottengano



# Livello di trasporto

Application	Data loss	Bandwidth	Time Sensitive
file transfer	no loss	elastic	no
e-mail	no loss	elastic	no
Web documents	no loss	elastic	no
real-time audio/video	loss-tolerant	audio: 5kbps-1Mbps video: 10kbps-5Mbps	yes, 100's msec
stored audio/video	loss-tolerant	same as above	yes, few secs
interactive games	loss-tolerant	few kbps up	yes, 100's msec
instant messaging	no loss	elastic	yes and no



# Protocolli del livello di trasporto

## ▶ TCP

- ▶ **Orientato alla connessione:** richiesto setup tra processo client e server
- ▶ **Trasporto affidabile** tra processo mittente e destinatario
- ▶ **Controllo del flusso:** mittente non inonderà il ricevente
- ▶ **Congestion control:** mittente considera stato della rete quando deve inviare nuovi messaggi
- ▶ **Non fornisce:** timing, garanzie su banda minima



# Protocolli del livello di trasporto

## ▶ UDP

- ▶ **Trasferimento dati non affidabile** tra processo mittente e destinatario
- ▶ **Non fornisce:** setup della connessione, affidabilità, controllo del flusso, controllo della congestion, timing e garanzie sulla banda



# Protocolli del livello di trasporto

<b>Application</b>	<b>Application layer protocol</b>	<b>Underlying transport protocol</b>
e-mail	SMTP [RFC 2821]	TCP
remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
file transfer	FTP [RFC 959]	TCP
streaming multimedia	proprietary (e.g. RealNetworks)	TCP or UDP
Internet telephony	proprietary (e.g., Dialpad)	typically UDP

