SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Lezione 3
Introduzione allo stack protocollare TCP/IP

Prof.ssa Gaia Maselli
maselli@di.uniroma1.it

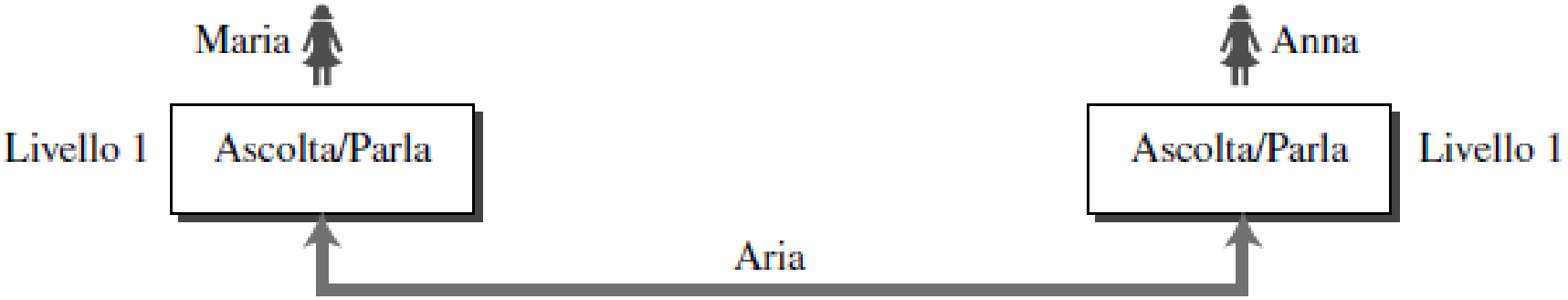
Hardware e software

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Si è fornita una panoramica della struttura e delle prestazioni di Internet, che è costituita da numerose reti di varie dimensioni interconnesse tramite opportuni dispositivi di comunicazione.
- Tuttavia per poter comunicare non è sufficiente assicurare questi collegamenti, ma è necessario utilizzare sia dell'hardware che del software!
- Hardware e software devono essere coordinati

Esempio di comunicazione

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



- Due interlocutori rispettano un protocollo di conversazione (interazione)
- Si inizia con un saluto
- Si adotta un linguaggio appropriato al livello di conoscenza
- Si tace mentre l'altro parla
- La conversazione si sviluppa come dialogo piuttosto che un monologo
- Si termina con un saluto

Protocollo

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Un protocollo definisce le regole che il mittente e il destinatario, così come tutti i sistemi intermedi coinvolti, devono rispettare per essere in grado di comunicare.

- In situazioni particolarmente semplici potrebbe essere sufficiente un solo protocollo, in situazioni più complesse potrebbe essere opportuno suddividere i compiti fra più livelli (layer), nel qual caso è richiesto un protocollo per ciascun livello: si parla dunque di layering di protocolli.

Organizzazione a più livelli

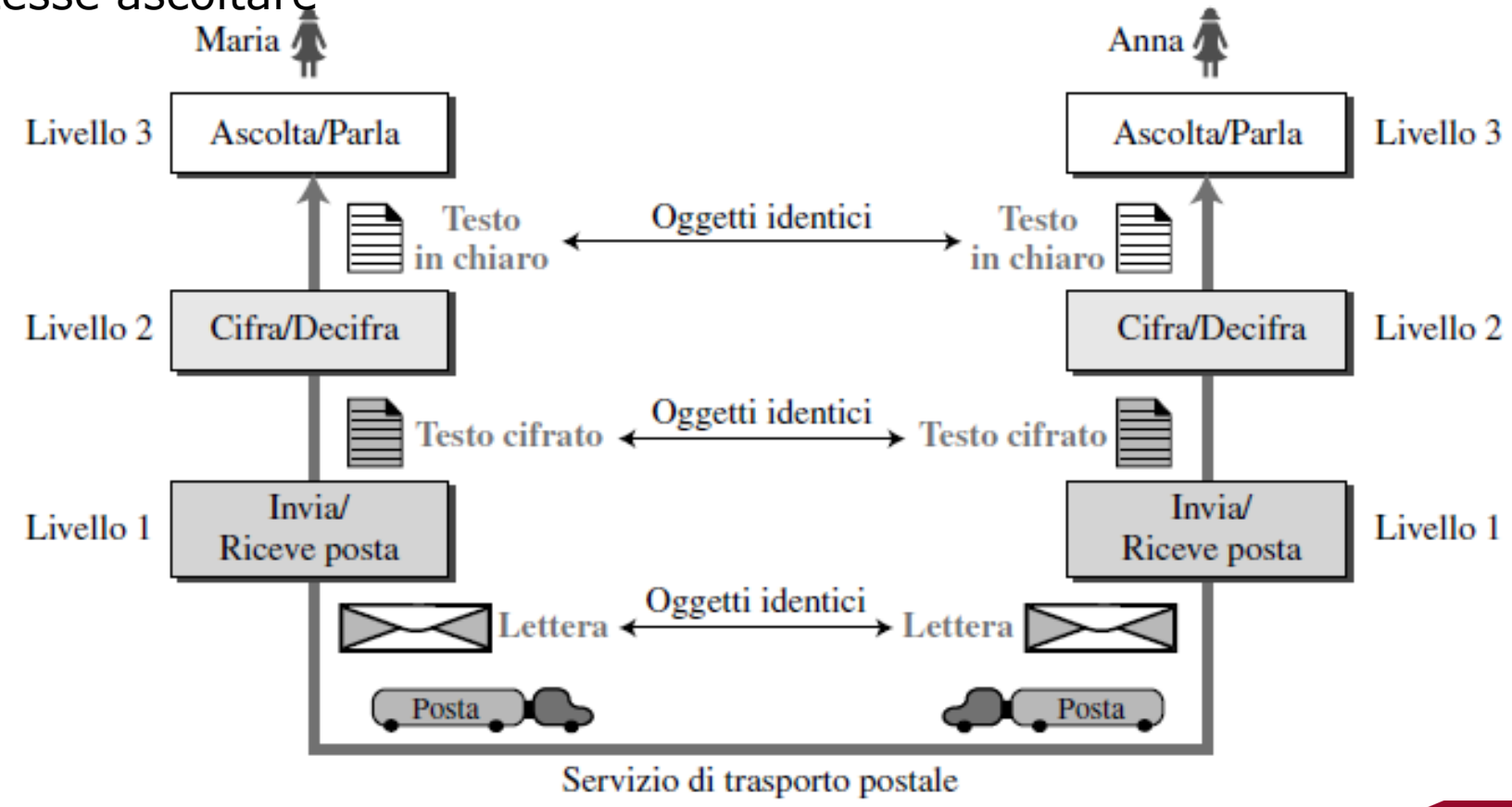
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Anna viene trasferita e le due amiche continuano a sentirsi via posta.
- Poiché hanno in mente un progetto innovativo vogliono rendere sicura la conversazione mediante un meccanismo di crittografia
- Il mittente di una lettera la cripta per renderla incomprensibile a un eventuale intruso, il destinatario la decripta per recuperare il messaggio originale

Esempio: Organizzazione a tre livelli

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Si ipotizza che le due amiche abbiano tre macchine ciascuna per portare a termine i compiti di ciascun livello
- Supponiamo che Maria invii la prima lettera.
- Maria comunica con la macchina al terzo livello come se fosse Anna e la potesse ascoltare



Strutturazione a livelli

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

La strutturazione dei protocolli in livelli consente di suddividere un compito complesso in compiti più semplici

Si potrebbe usare una sola macchina ma cosa accadrebbe se le due amiche decidessero di cambiare tecnica di crittografia?

- Nel caso delle 3 macchine verrebbe sostituita solo quella intermedia

  ➔ *Modularizzazione* (indipendenza dei livelli)
- Un modulo (livello) può essere considerato come un black box con opportuni ingressi e uscite, senza preoccuparsi delle modalità con cui i dati in ingresso vengano trasformati nei dati di uscita
- Se due macchine forniscono lo stesso output dato il medesimo input allora possono essere considerate equivalenti ➔ Le macchine possono essere acquistate da fornitori diversi
- Separazione tra servizi e implementazione: un livello usa servizi dal livello inferiore e offre servizi al livello superiore – indipendentemente da come sia implementato

Principi della strutturazione a livelli

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

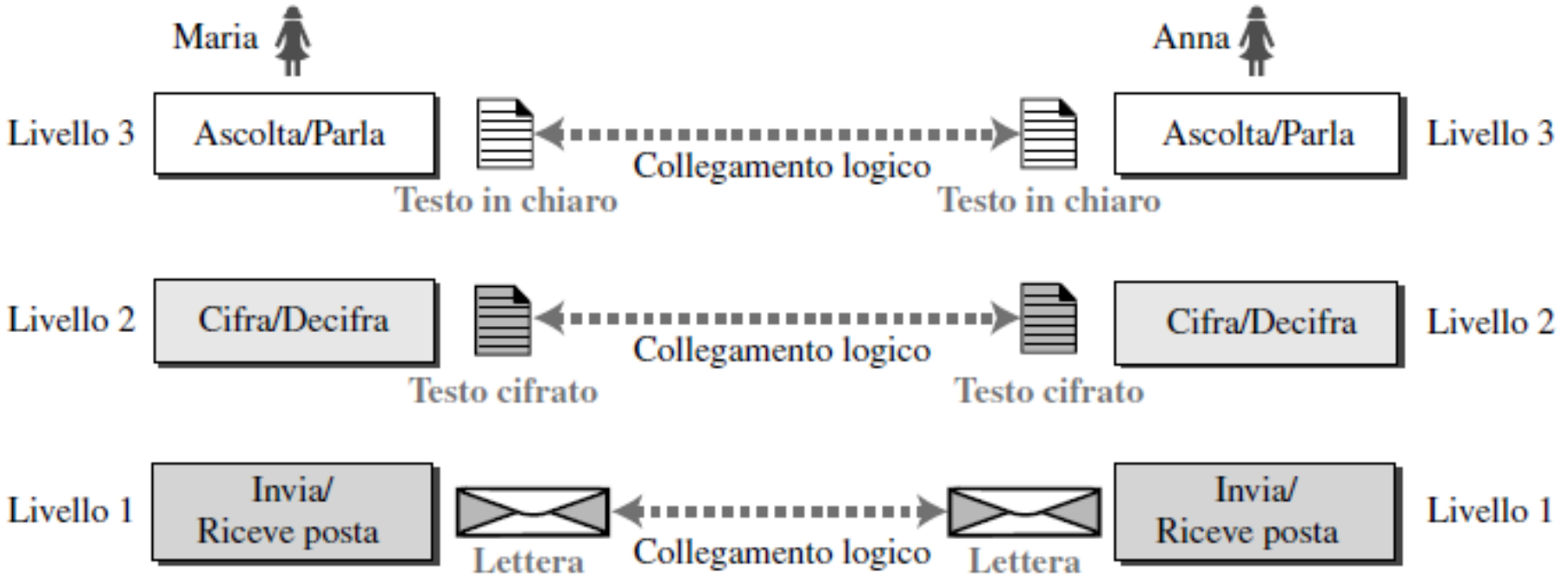
Quando è richiesta una comunicazione ***bidirezionale***, ciascun livello deve essere capace di effettuare i due compiti opposti, uno per ciascuna direzione (es., crittografare, decrittografare)

Gli oggetti in ***input/output*** sotto ciascun livello di entrambi i lati devono essere ***identici*** (es. sotto il livello 2 c'è una una lettera cifrata)

Collegamento logico fra i livelli

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Collegamento logico: i livelli logicamente sono direttamente collegati, ovvero il protocollo implementato a ciascun livello specifica una comunicazione diretta fra i pari livelli delle due parti



E in Internet?

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Quali compiti esegue la rete?

Lo stack protocollare TCP/IP

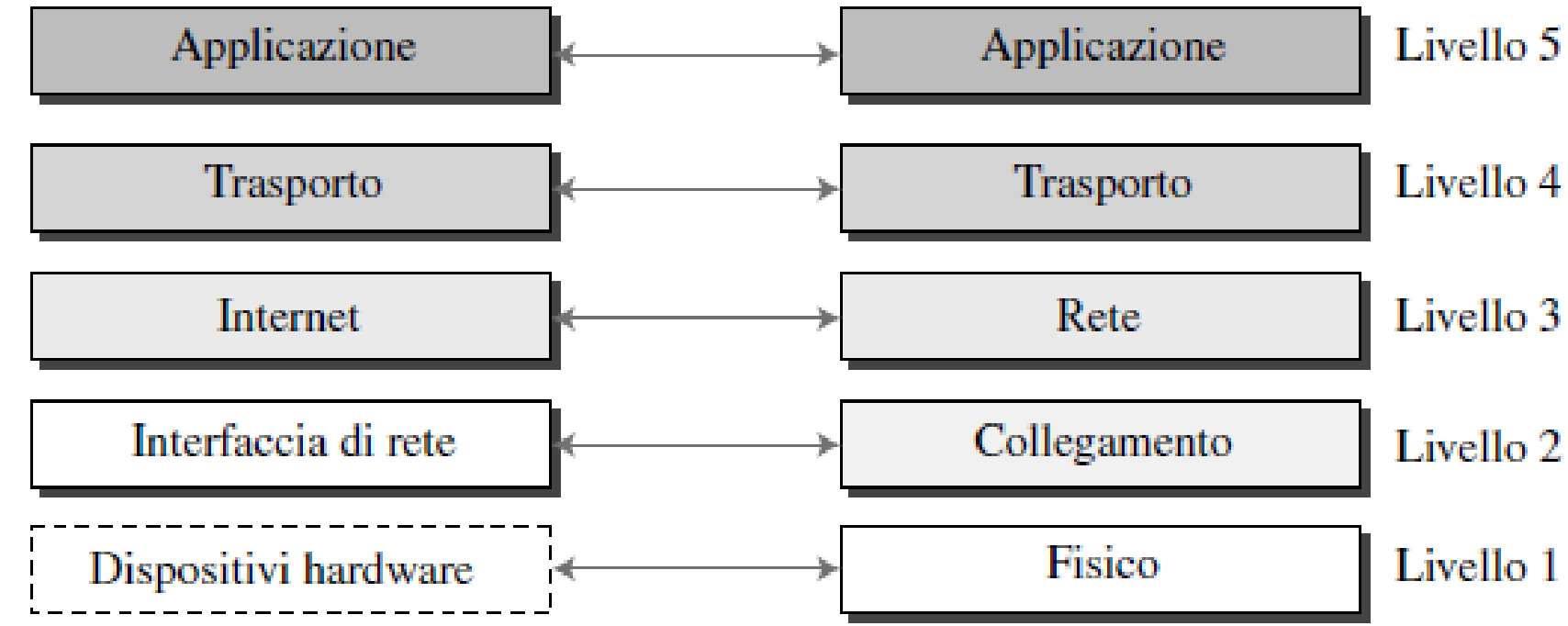
TCP/IP è una famiglia di protocolli attualmente utilizzata in Internet. Si tratta di una gerarchia di protocolli costituita da moduli interagenti, ciascuno dei quali fornisce funzionalità specifiche.

Il termine **gerarchia** significa che ciascun protocollo di livello superiore è supportato dai servizi forniti dai protocolli di livello inferiore.

Definita in origine in termini di quattro livelli software soprastanti a un livello hardware, la pila TCP/IP è oggi intesa come composta di cinque livelli.

I livelli nello stack protocollare TCP/IP

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



a. Livelli nel modello obsoleto

b. Livelli nel modello attuale utilizzato in questo volume

Pila di protocolli TCP/IP (stack protocollare, protocol stack)

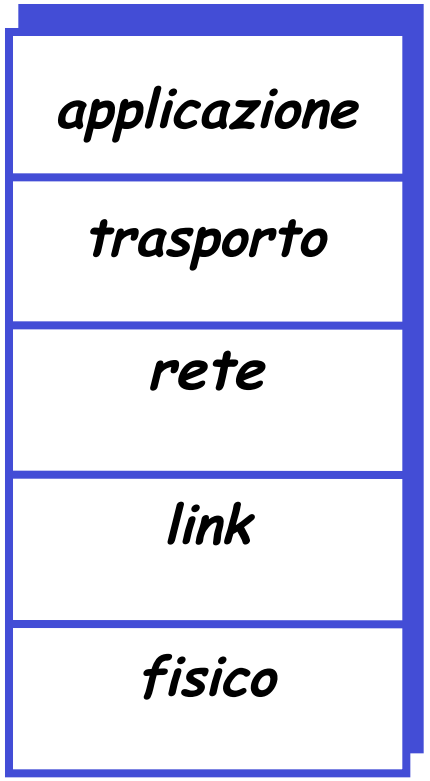
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Applicazione: è la sede delle applicazioni di rete

- HTTP, SMTP, FTP, DNS
- I pacchetti sono denominati ***messaggi***

Trasporto: trasferimento dei messaggi a livello di applicazione tra il modulo client e server di un'applicazione

- TCP, UDP
- I pacchetti sono denominati ***segmenti***



Pila di protocolli TCP/IP

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

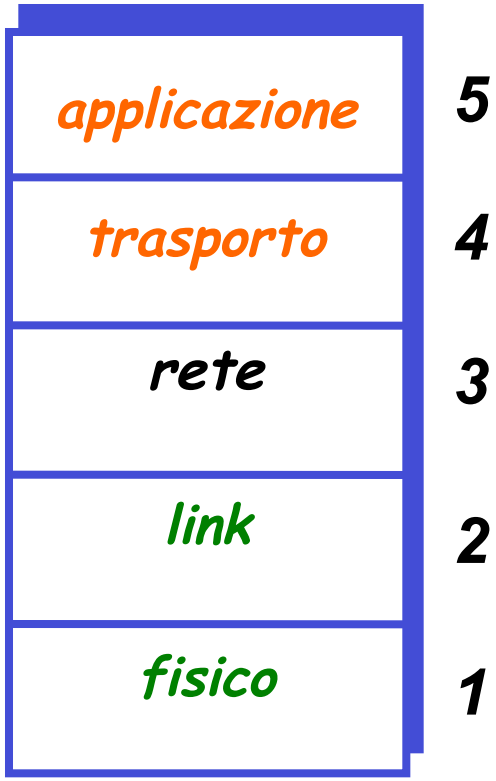
Rete: instradamento dei segmenti dall'origine alla destinazione

- IP, protocolli di instradamento
- I pacchetti sono denominati ***datagrammi***

Link (collegamento): trasmettere datagrammi da un nodo a quello successivo sul percorso

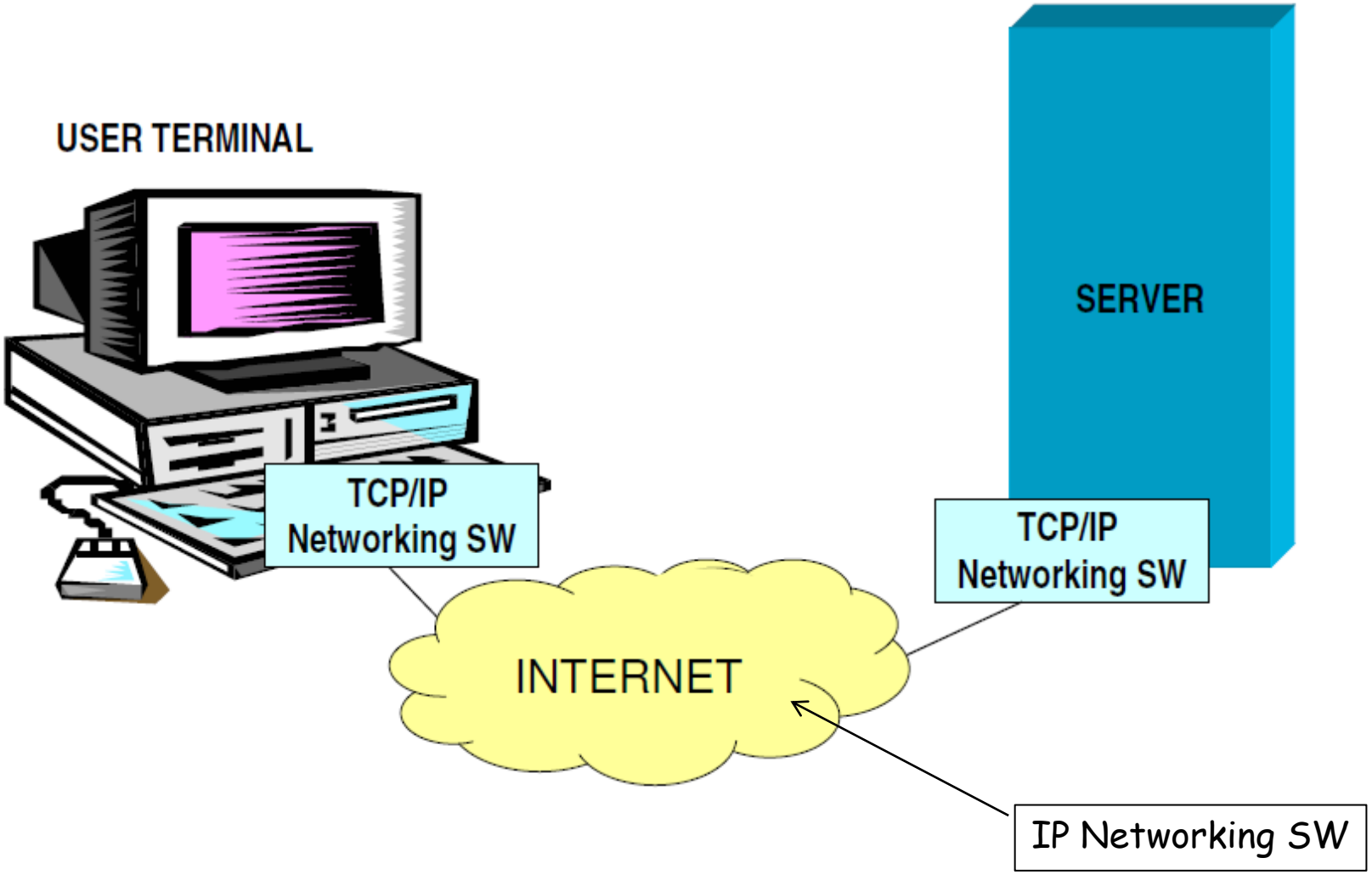
- Ethernet, Wi-Fi, PPP
- Lungo un percorso sorgente-destinazione un datagramma può essere gestito da protocolli diversi
- I pacchetti sono denominati ***frame***

fisico: trasferimento dei singoli bit



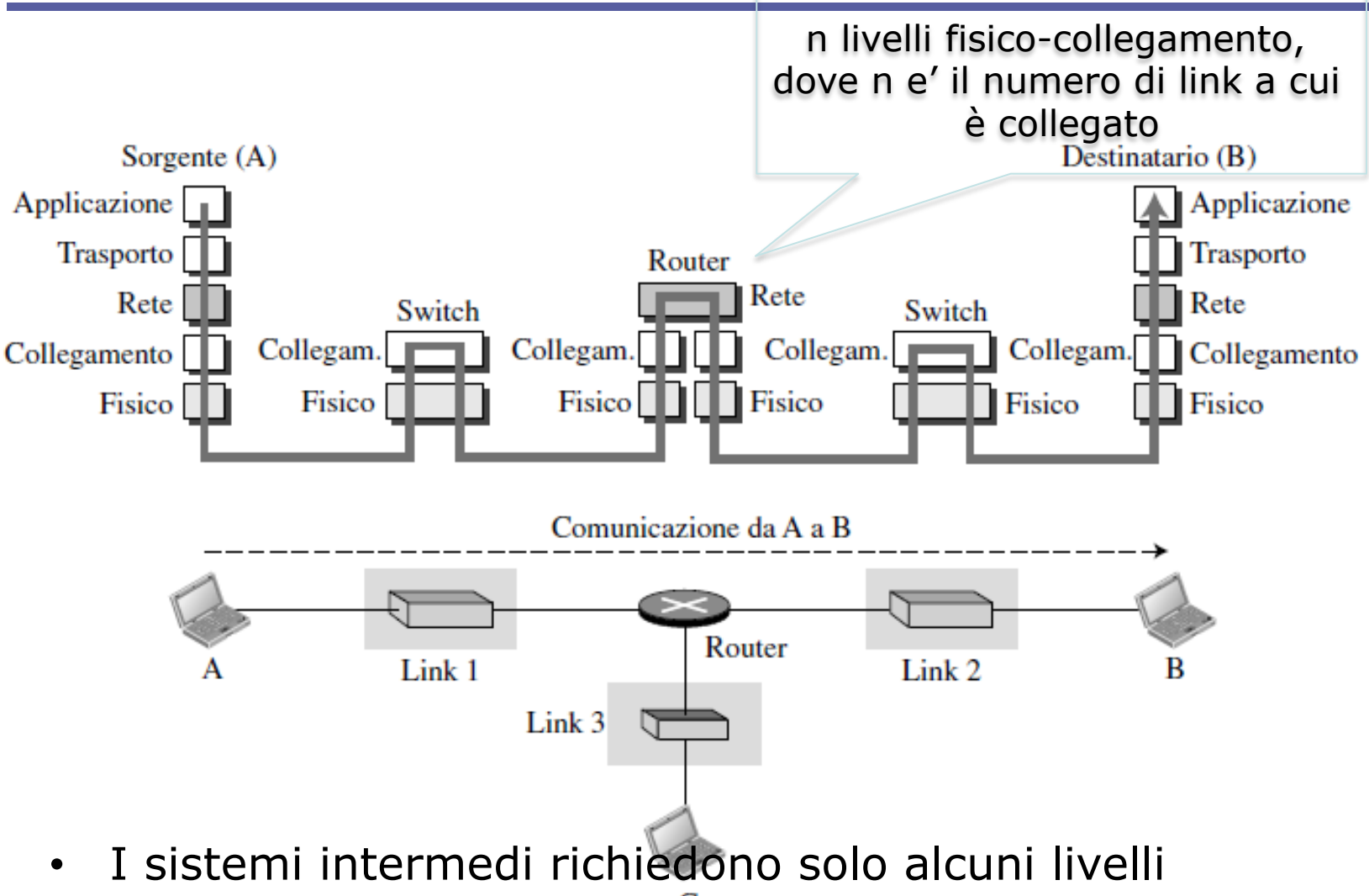
Dove si trova il software di rete?

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



Comunicazione in una internet

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



- I sistemi intermedi richiedono solo alcuni livelli
- Grazie al layering tali sistemi implementano solo i livelli necessari, riducendo la complessità

Gerarchia di protocolli

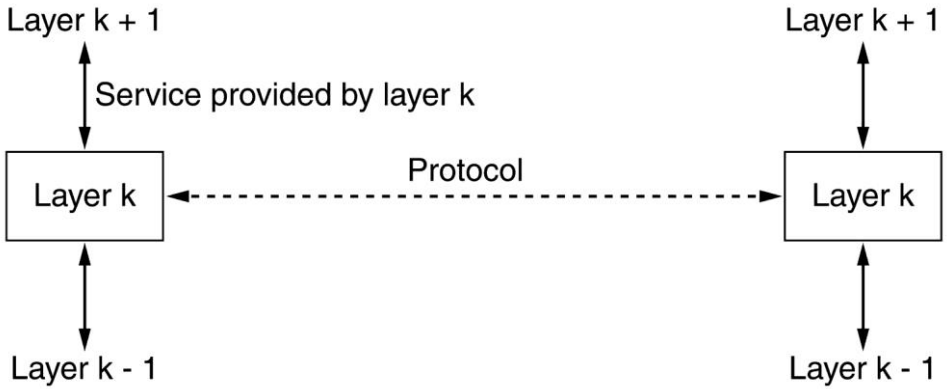
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- La rete è organizzata come pila di **strati** (layer) o **livelli**, costruiti l'uno sull'altro
- Lo scopo di ogni strato è quello di **offrire** determinati **servizi** agli strati di livello superiore, nascondendo i dettagli di implementazione
- Lo strato N di un computer è in comunicazione con lo strato N di un altro computer
- Le regole e le convenzioni usate in questa comunicazione sono globalmente note come i **protocolli** dello strato N
- Le entità che formano gli strati sono chiamati **pari** (peer)
- I pari comunicano usando il protocollo
- I dati non sono trasferiti direttamente dallo strato N di un computer allo strato N di un altro computer !!!

Servizi e protocolli

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

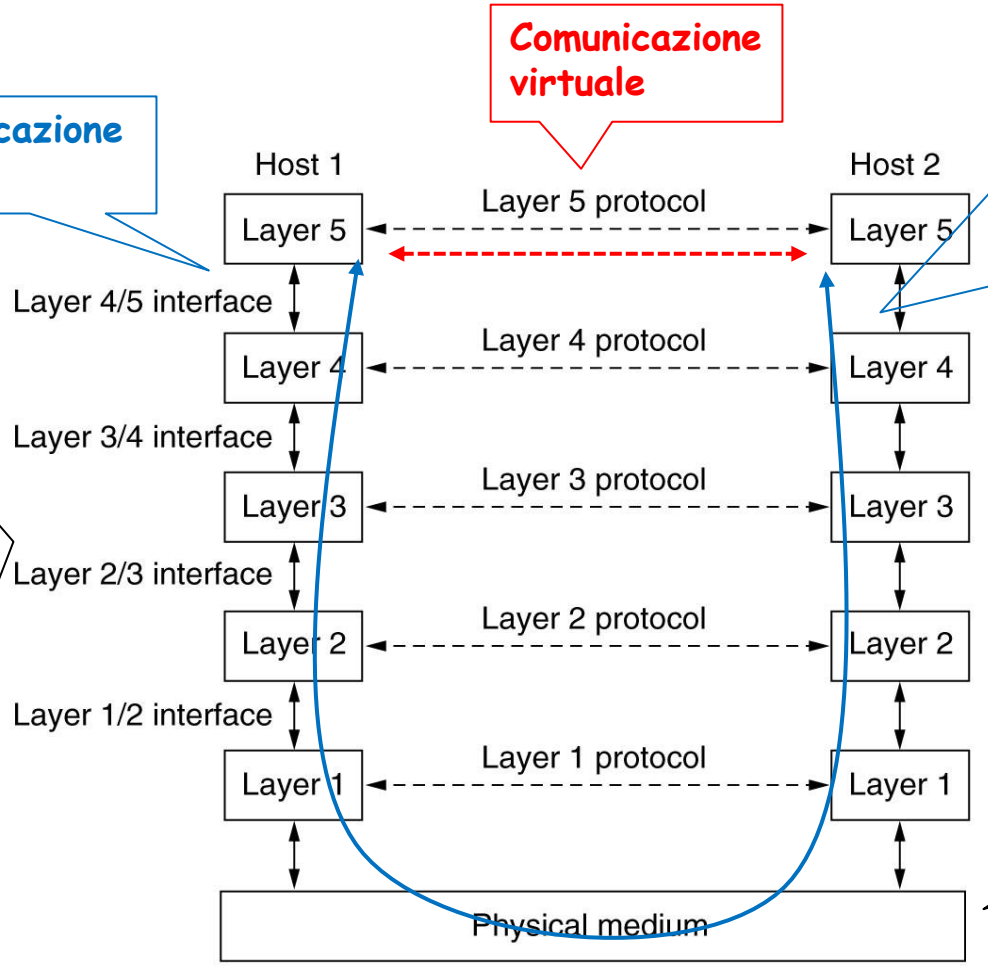
- Sono concetti ben distinti
- Un servizio è un insieme di primitive che uno strato offre a quello superiore
  - Definisce quali operazioni lo strato è in grado di offrire, ma non dice nulla di come queste operazioni sono implementate
  - E' correlato all'interfaccia tra due strati, dove quello inferiore è il provider del servizio, mentre quello superiore è l'utente
- Un protocollo è un insieme di regole che controllano il formato e il significato dei pacchetti, o messaggi scambiati tra le entità pari all'interno di uno strato



Architettura di rete

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Ogni strato passa dati e informazioni di controllo allo strato immediatamente **sottostante**, fino a raggiungere quello più basso



ESEMPIO:
I processi dello strato 4 modellano la loro comunicazione come se fosse "orizzontale":
- `SendToOtherSide`
- `GetFromOtherSide`

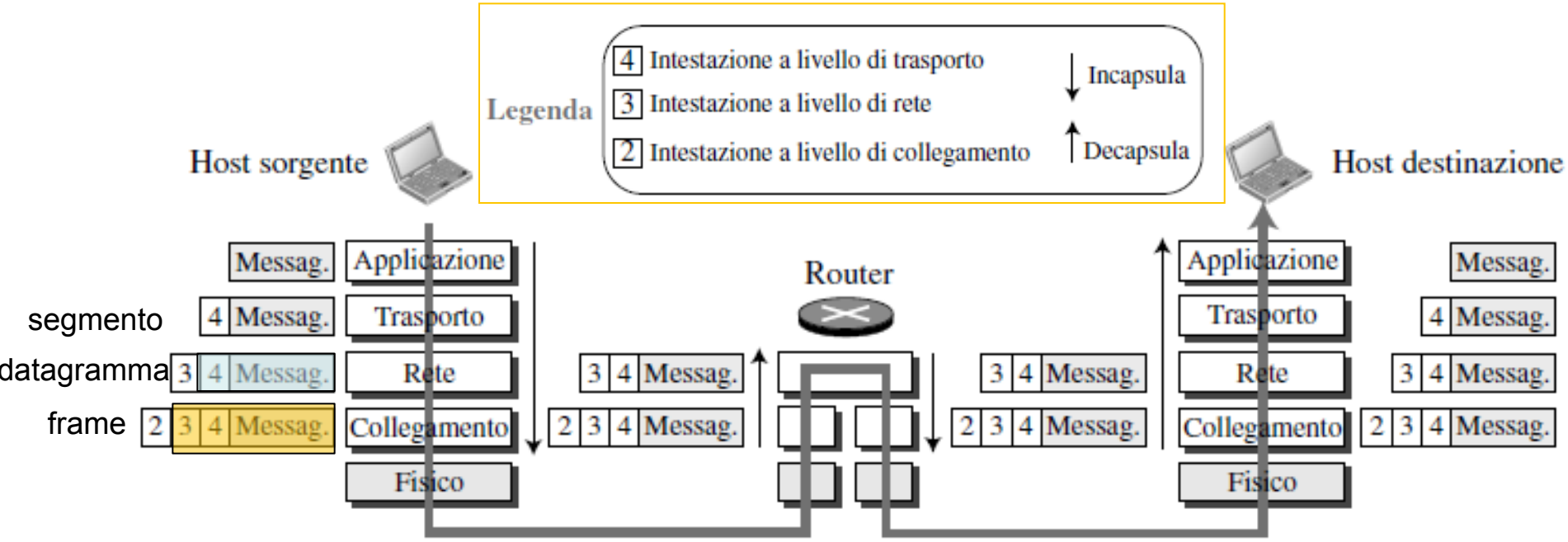
In realtà queste procedure comunicano mediante strati inferiori

Interfacce: definiscono le operazioni elementari e i servizi che lo strato inferiore rende disponibili a quello soprastante

livello **fisico** attraverso cui avviene la comunicazione vera e propria

Incapsulamento e decapsulamento

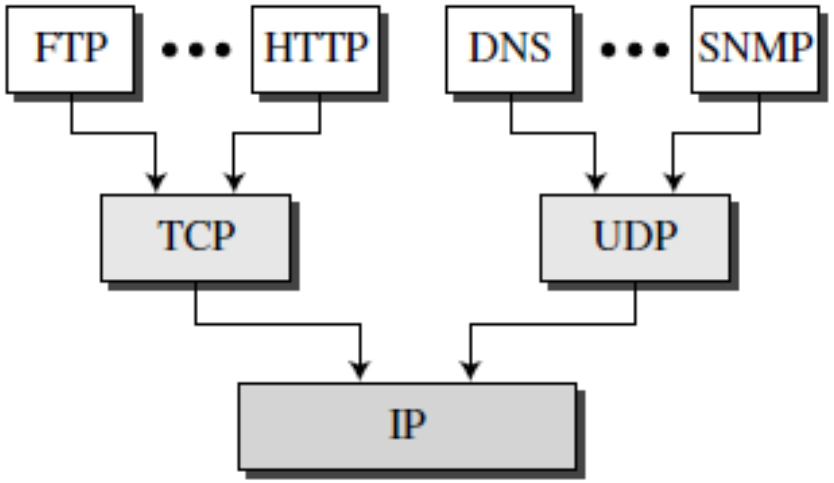
- La sorgente effettua l'incapsulamento (prende il pacchetto dal livello superiore, lo considera come carico dati o payload e aggiunge un header o intestazione).
  - Messaggio (nessuna intestazione)
  - Segmento o datagramma utente = header trasporto + messaggio
  - Datagramma = header rete + segmento
  - Frame = header collegamento + datagramma
- Il destinatario effettua il decapsulamento
- Il router effettua sia incapsulamento che decapsulamento perché collegato a due link



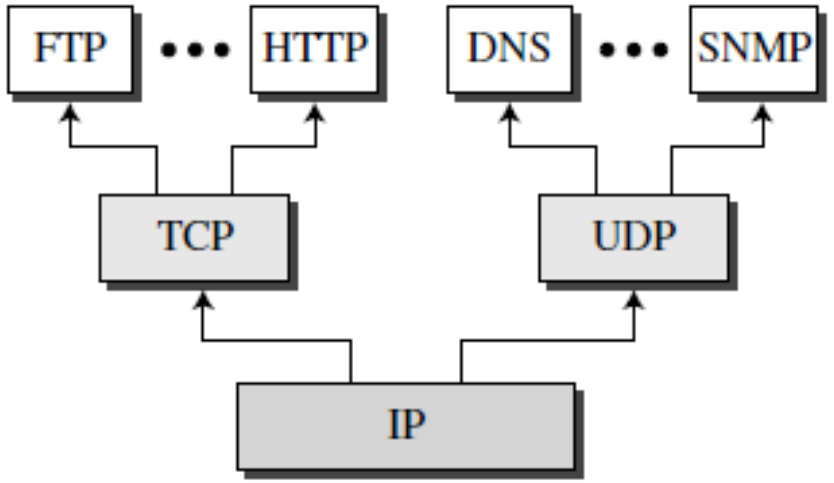
Multiplexing e demultiplexing

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Dato che lo stack protocollare TCP/IP prevede più protocolli nello stesso livello, è necessario eseguire il multiplexing alla sorgente e il demultiplexing alla destinazione
- Multiplexing: un protocollo può incapsulare (uno alla volta) i pacchetti ottenuti da più protocolli del livello superiore
- Demultiplexing: un protocollo può decapsulare e consegnare i pacchetti a più protocolli del livello superiore
- E` necessario un campo nel proprio header per identificare a quale protocollo appartengano i pacchetti incapsulati



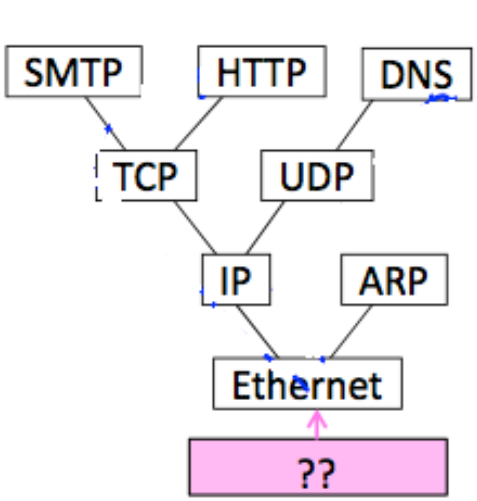
a. Multiplexing alla sorgente

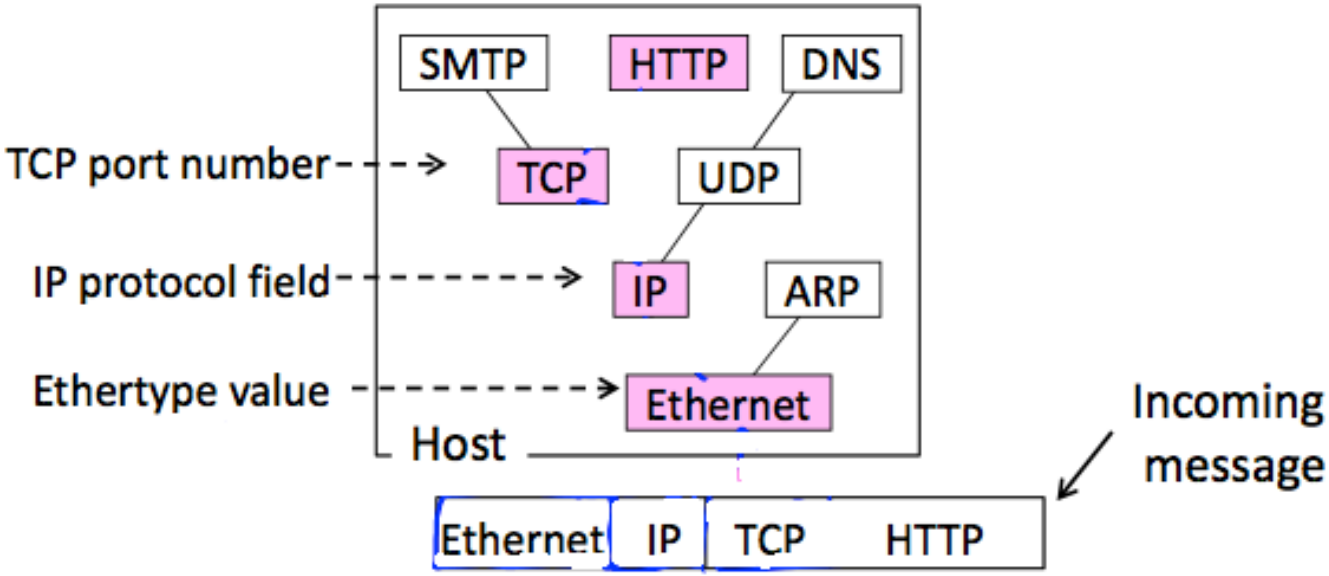


b. Demultiplexing alla destinazione

Demultiplexing

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



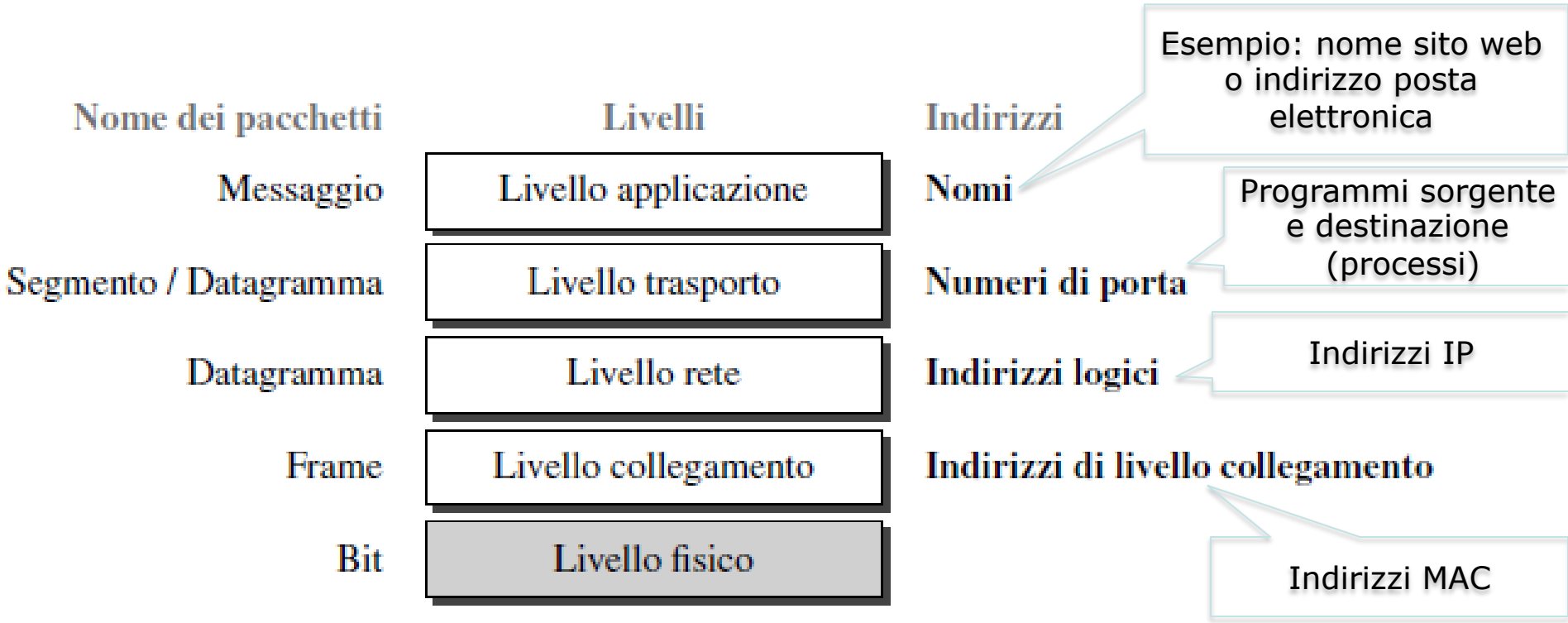


- Per poter effettuare le operazioni di multiplexing e demultiplexing, ogni pacchetto deve avere un campo all'interno dell'header per identificare a quale protocollo appartiene

Indirizzamento nel modello TCP/IP

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Poiché il modello TCP/IP prevede una comunicazione logica tra coppie di livelli è necessario avere un indirizzo sorgente e un indirizzo destinazione ad ogni livello



Layering: vantaggi

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

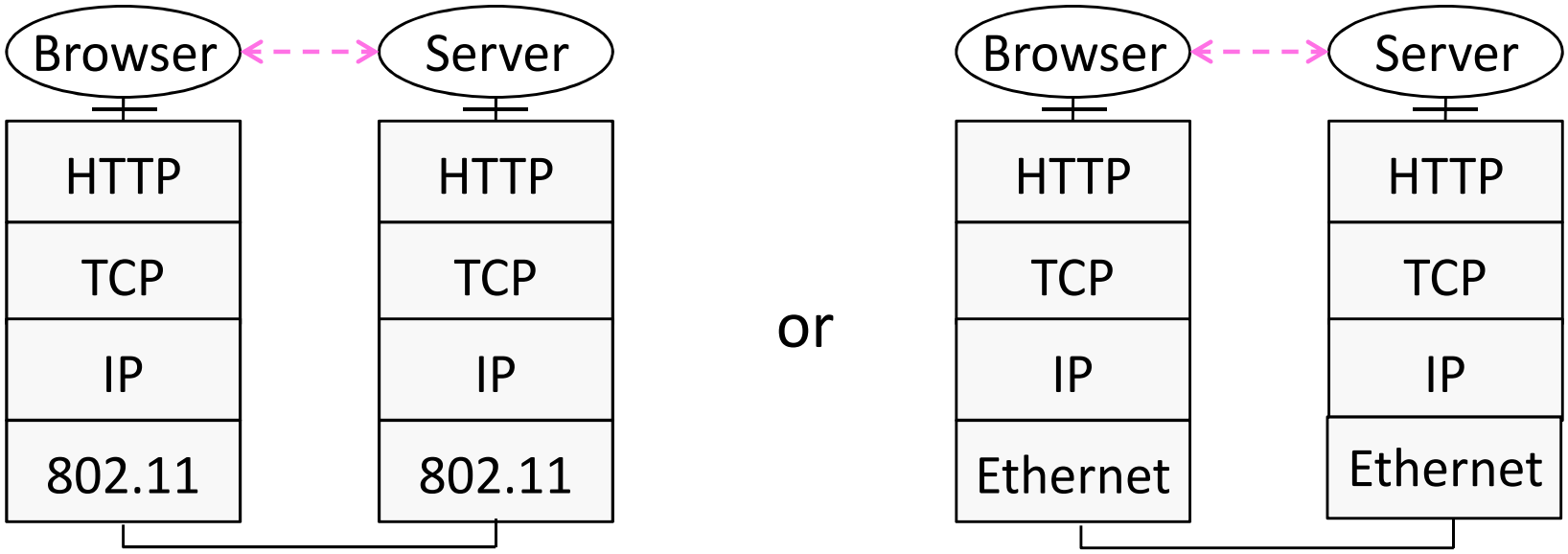
Modularità

- Semplicità di design
- Possibilità di modificare un modulo in modo trasparente se le interfacce con gli altri livelli rimangono le stesse
- Possibilità per ciascun costruttore di adottare la propria implementazione di un livello purchè requisiti su interfacce soddisfatti

Layering: vantaggi

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Riuso per gestire eterogeinità



Layering: svantaggi

- A volte necessario scambio di informazioni tra livelli non adiacenti (esempio: per ottimizzare app funzionante su wireless) non rispettando principio della stratificazione

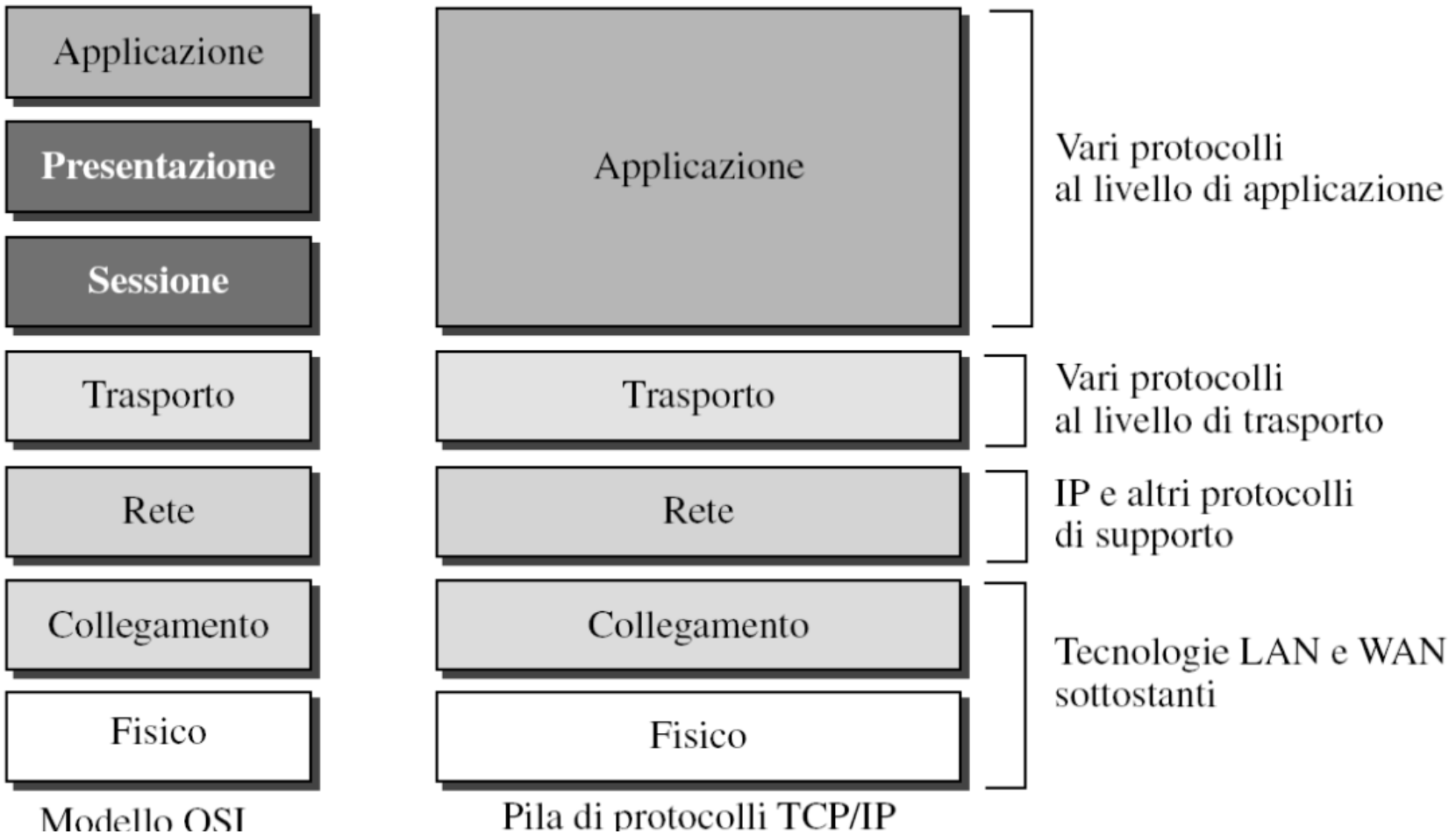
Modello OSI

L'ISO (International Organization for Standardization) – organizzazione dedicata alla definizione di standard universalmente accettati – ha definito il modello OSI (Open System Interconnection) come modello alternativo al TCP/IP

- Framework stratificato per il progetto di sistemi di rete che consentono la comunicazione fra qualsiasi tipo di dispositivo

Confronto tra OSI e TCP/IP

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



Insuccesso del modello OSI

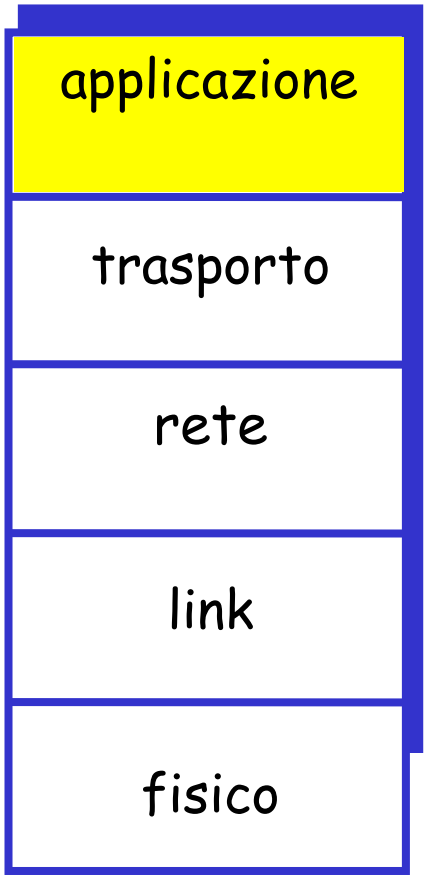
- L'OSI venne pubblicato quando il TCP/IP era già ampiamente diffuso e gli erano state dedicate parecchie risorse: un'eventuale sostituzione avrebbe comportato un costo notevole
- Alcuni livelli, come presentazione e sessione non sono mai stati completamente specificati (software corrispondente mai stato completamente sviluppato)
- Non riuscirono a dimostrare delle prestazioni tali da convincere le autorità di Internet a sostituire il TCP/IP

MODELLO TCP/IP

Livello applicazione

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Principi delle applicazioni di rete
- Web e HTTP
- FTP
- Posta elettronica
  - SMTP, POP3, IMAP
- DNS
- Applicazioni P2P
- Programmazione delle socket con TCP
- Programmazione delle socket con UDP



Introduzione

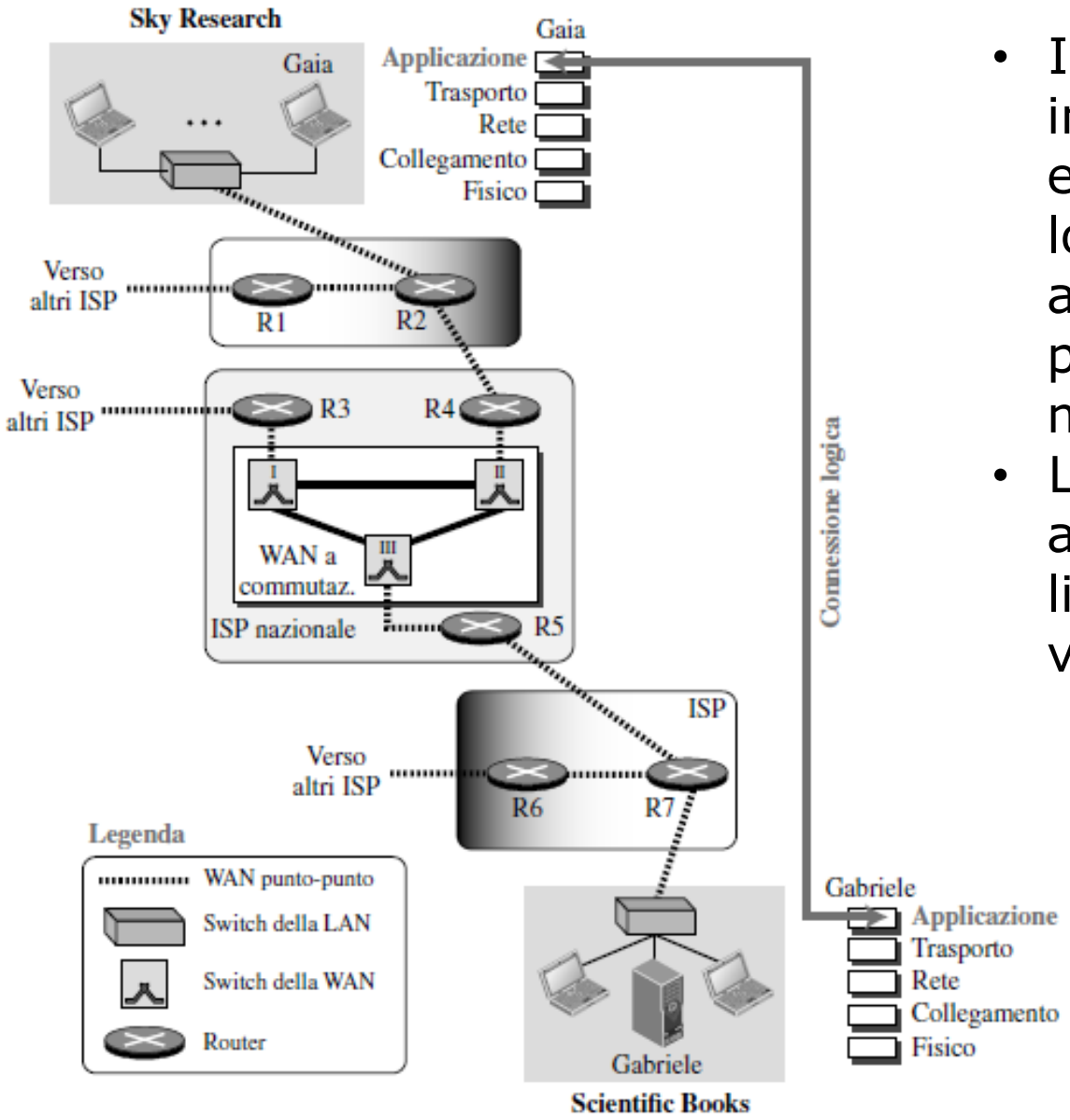
SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Il livello applicazione fornisce servizi all'utente.

La comunicazione è fornita per mezzo di una connessione logica: questo significa che i livelli applicazione nei due lati della comunicazione agiscono come se esistesse un collegamento diretto attraverso il quale poter inviare e ricevere messaggi.

Connessione logica a livello applicazione

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA



- I due utenti possono immaginare che tra di essi esista un canale logico bidirezionale attraverso il quale si possono inviare messaggi
- La comunicazione reale avviene attraverso più livelli e più dispositivi, e vari canali fisici

L’offerta di servizi

- Internet è stata progettata per fornire servizi agli utenti. Dato che il livello applicazione è l’unico che fornisce servizi agli utenti di Internet, la sua flessibilità consente di aggiungere nuovi protocolli con estrema facilità, così come si è verificato nella storia di Internet e sta tuttora avvenendo.
- Alla nascita di Internet solo alcuni protocolli di livello applicazione erano disponibili per gli utenti; oggi non è più possibile indicare il numero dei protocolli esistenti poiché ne vengono costantemente aggiunti di nuovi.
- Un protocollo che viene aggiunto a un dato livello deve essere progettato in modo da usare i servizi del livello inferiore
- Aggiungere o eliminare protocolli dal livello applicazione è relativamente facile perché non comporta modifiche agli altri livelli

L'offerta di servizi

- Protocolli standard
  - Esistono diversi protocolli di livello applicazione che sono standardizzati e documentati dagli enti responsabili della gestione di Internet
  - Ogni protocollo standard è costituito da una coppia di programmi che interagiscono con l'utente e con il livello di trasporto per fornire uno specifico servizio
  - Es. Applicazione Web specificata dal protocollo di comunicazione HTTP
- Protocolli non standard
  - E' possibile creare un'applicazione non standard scrivendo due programmi che forniscono servizi agli utenti, facendo uso dei servizi di trasporto
  - Non è necessario chiedere autorizzazioni: un'azienda può sviluppare il proprio protocollo di livello applicazione per far comunicare i propri uffici sparsi nel mondo

Alcune applicazioni di rete

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- Posta elettronica
- Web
- Messaggistica istantanea
- Autenticazione in un calcolatore remoto
- Condivisione di file P2P
- Giochi multiutente via rete
- Streaming di video-clip memorizzati
- Telefonia via Internet
- Videoconferenza in tempo reale

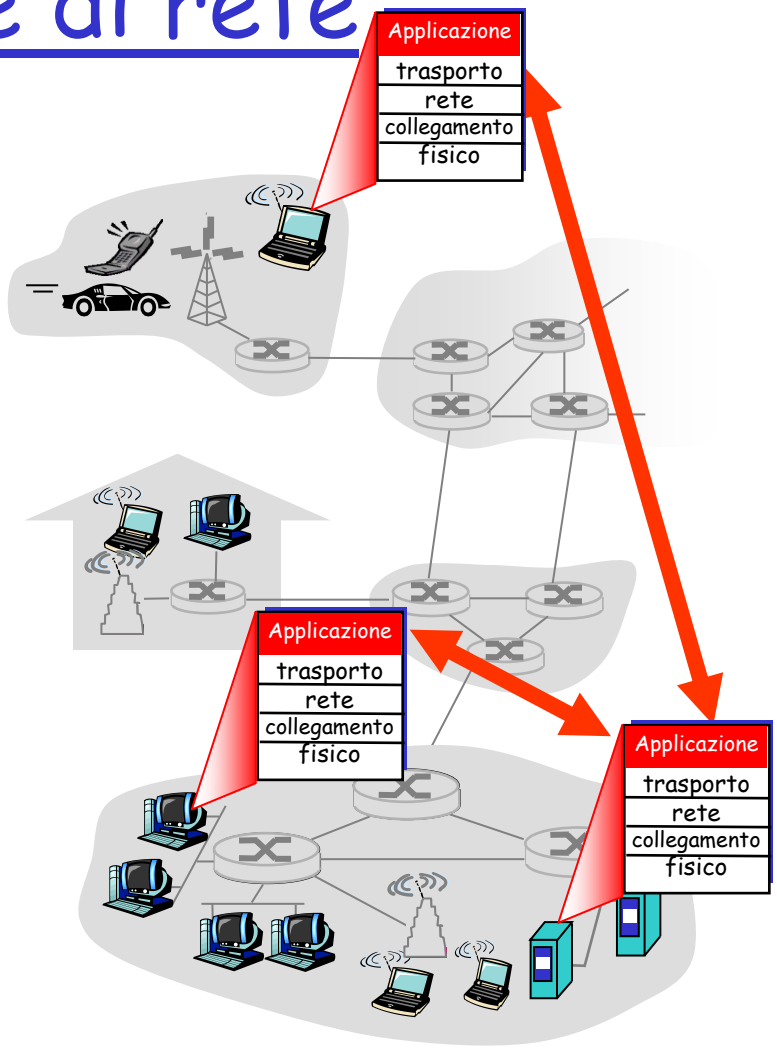
Creare un'applicazione di rete

Scrivere programmi che

- girano su sistemi terminali diversi
- comunicano attraverso la rete
- Ad es. il software di un server Web comunica con il software di un browser

software in grado di funzionare su più macchine

- non occorre predisporre programmi per i dispositivi del nucleo della rete, quali router o commutatori Ethernet
- I programmi applicativi sono indipendenti dalla tecnologia che c'è sotto



Esempio: email

Una nuova applicazione di rete

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

Supponiamo di voler creare una nuova applicazione di rete, allora è necessario avere un piano architetturale:

Che tipo di **architettura** si vuole creare (client-server, peer-to-peer)?

Come **comunicano** i processi dell'applicazione?

Che tipo di **servizi** (di rete) richiede l'applicazione (affidabilità, banda)?

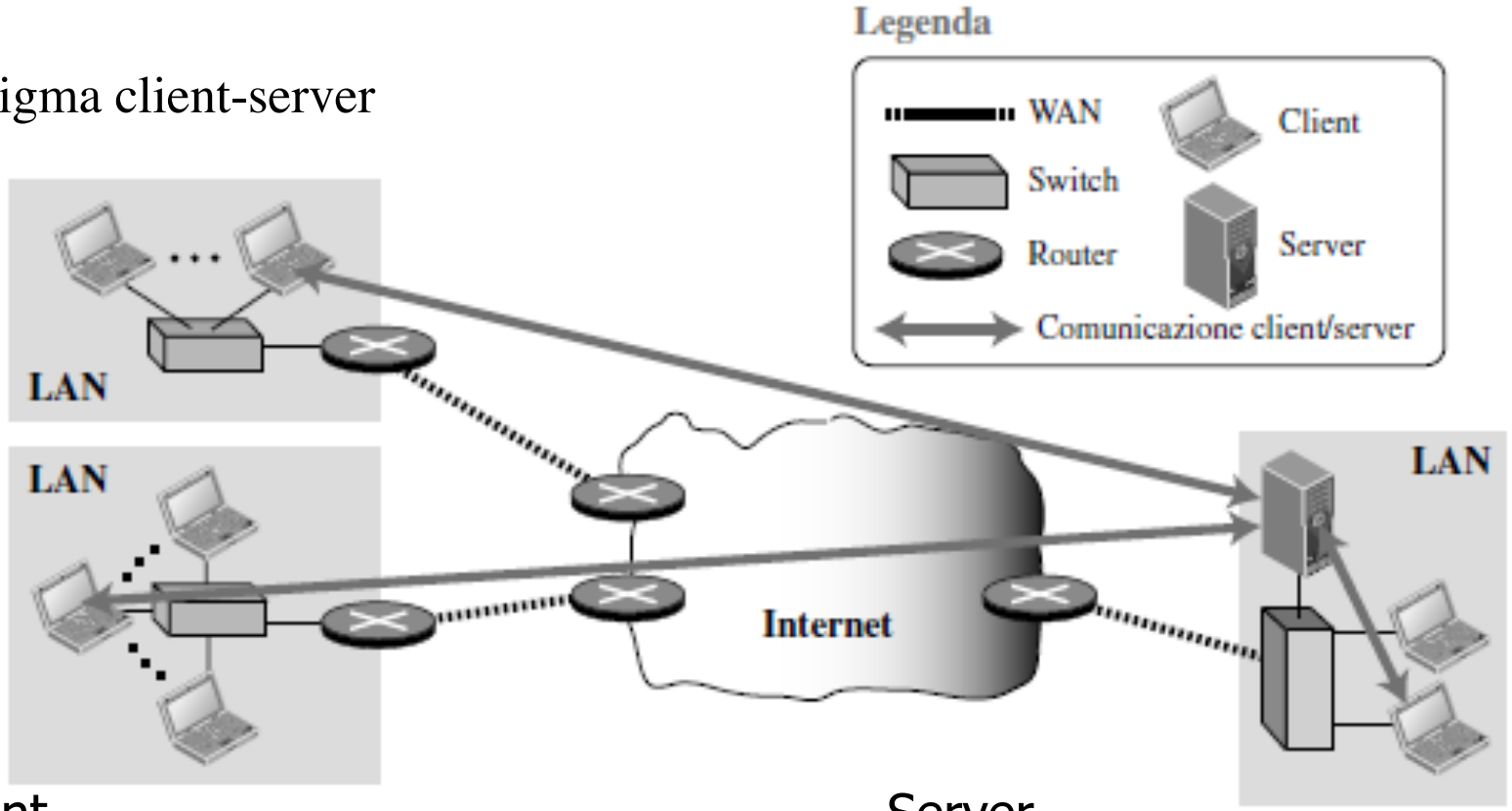
Architettura dell'applicazione

- I due programmi applicativi devono essere entrambi in grado di richiedere e offrire servizi, oppure ciascuno deve occuparsi di uno dei due compiti?

  Paradigma:

  - Client-server
  - Peer-to-peer (P2P)
  - Architetture ibride (client-server e P2P)

Paradigma client-server



- Client
  - Richiedente il servizio
  - In esecuzione solo quando è necessario il servizio
  - Numerosi client che richiedono il servizi
- Server
  - fornitore di servizi
  - Sempre in esecuzione, in attesa di richieste dal client
  - Numero limitato di processi server pronti a offrire uno specifico servizio

- Il ruolo delle due entità è totalmente differente: non è possibile eseguire un client come programma server e viceversa

Paradigma client/server

Svantaggi:

- Carico di comunicazione risulta concentrato sul server (che deve essere molto potente)
- Server farm per creare un potente server virtuale
- Costi di gestione per offrire servizio

Applicazioni tipiche

- WWW, posta elettronica, FTP, SSH