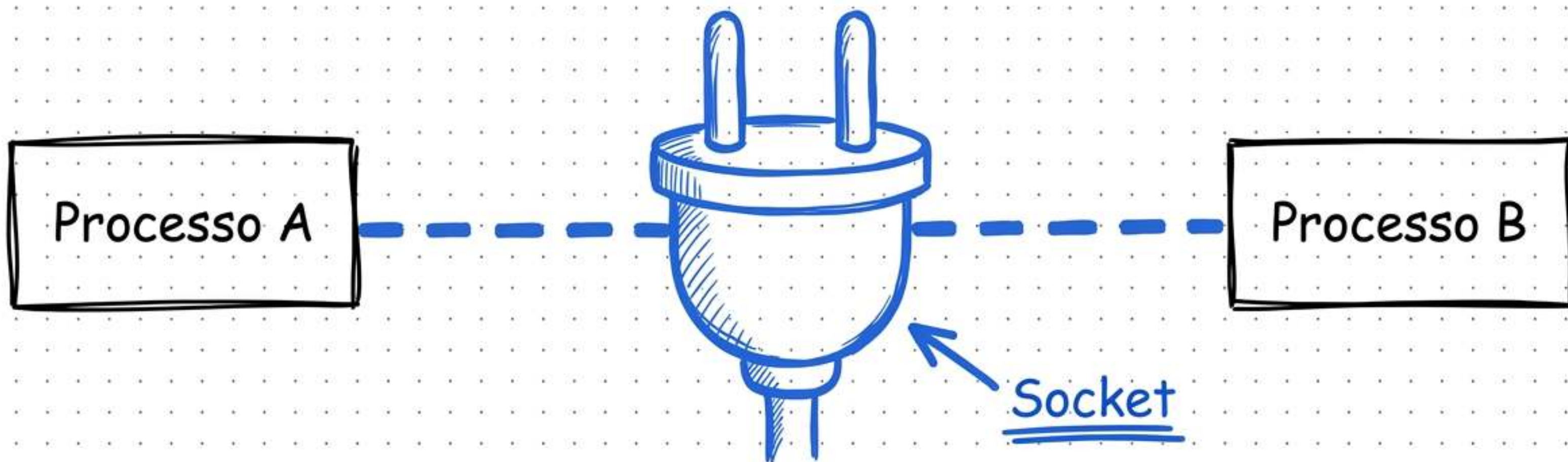


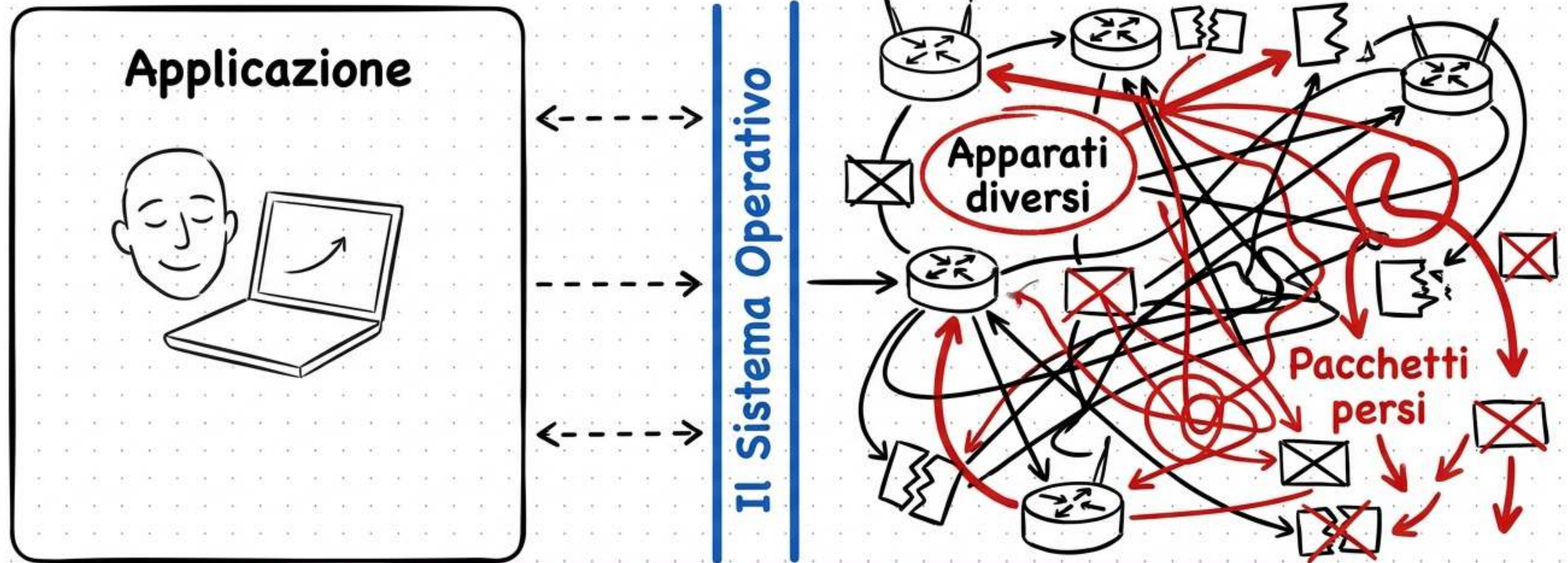
I Socket: Definizione e Funzionamento

Come le applicazioni comunicano in rete senza conoscere la rete.



Il Problema della Rete

Chi scrive un'applicazione non vuole gestire la frammentazione dei pacchetti, l'attraversamento dei router o la perdita dei dati.

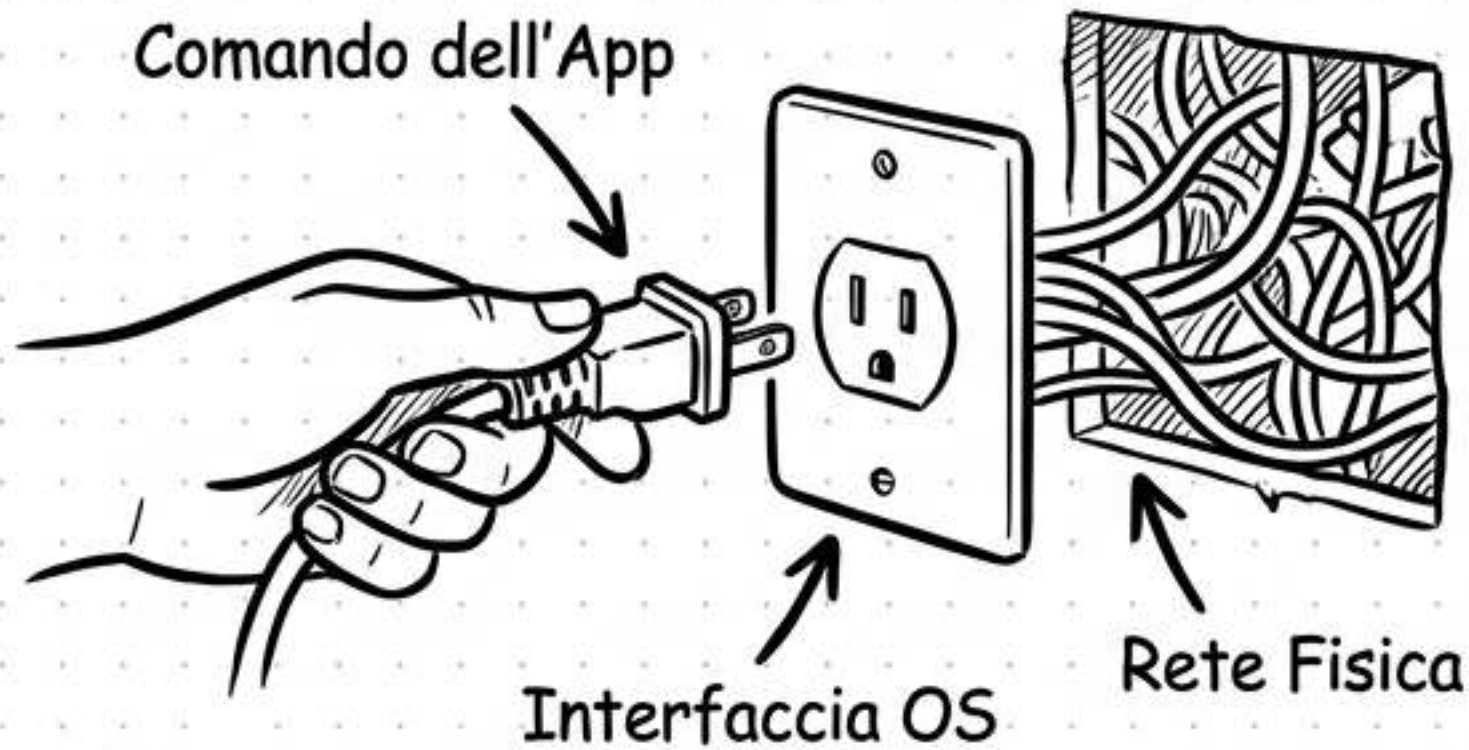


La soluzione? Il sistema operativo nasconde questa complessità dietro uno strumento già pronto.

L'Analogia della Presa Elettrica

Il termine inglese "socket" significa "presa". Non è il cavo e non è la corrente: è il punto di accesso normalizzato a un'infrastruttura esterna.

The Analogy



The Application Reality



Per il programmatore, usare un socket è esattamente come leggere o scrivere su un file. I dati viaggiano materialmente in modo invisibile.

Come viene identificato un socket?



Attenzione: Un singolo socket NON è l'intero collegamento,
è solo il punto terminale (endpoint).

La Regola d'Oro della Comunicazione

Per individuare una comunicazione completa servono due socket: quello locale e quello remoto.

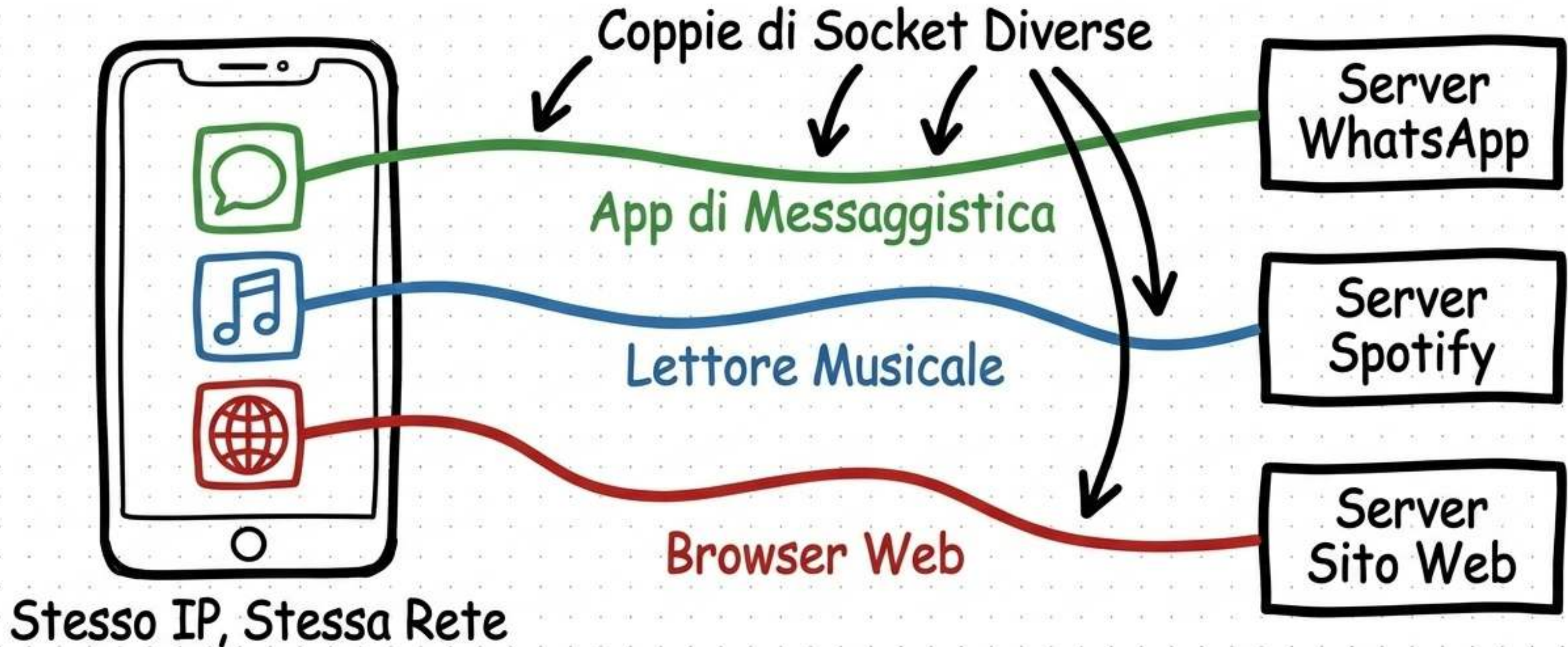


[Socket 1] + [Socket 2] = Conversazione Unica

È questa coppia di estremi a distinguere una conversazione da tutte le altre in corso sulla stessa macchina.

L'Esempio dello Smartphone

Come fa il sistema operativo a consegnare i dati giusti all'app giusta?
Ogni app usa una coppia di socket distinta.



Ruoli a Confronto: Client vs. Server

IL SERVER (Passivo)

- Crea un socket e lo associa a una porta nota (per convenzione).
- Lo mette in ascolto (non parla con uno specifico interlocutore, attende chiunque).

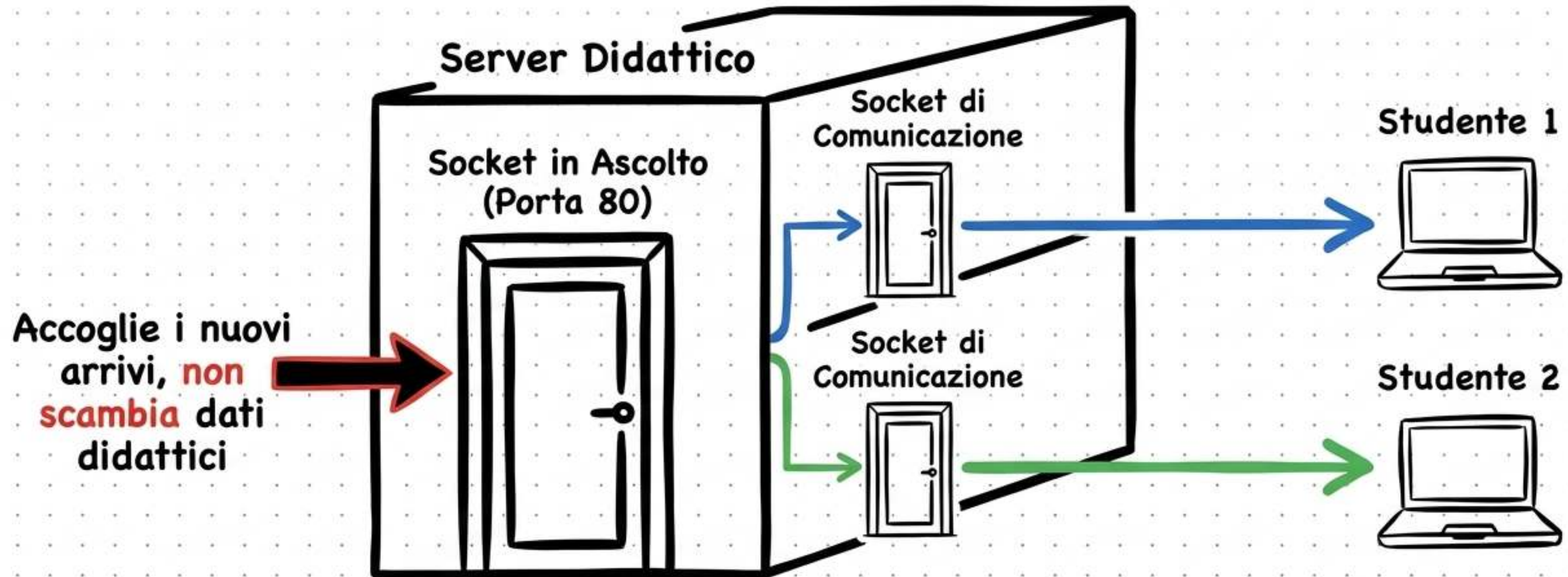
IL CLIENT (Attivo)

- Crea un socket per avviare il contatto verso il server.
- Usa una porta dinamica (assegnata dal sistema operativo al momento, non deve essere nota).

Il client non ha bisogno di una porta fissa e conosciuta; il server assolutamente sì!

Il Concetto di “Ascolto”

Quando 20 studenti si collegano al server, il socket in ascolto fa solo da usciere. Per scambiare dati con ogni studente, il **server** crea nuovi socket distinti.



Le Due Famiglie di Socket

Orientati alla Connessione



- **Caratteristiche:** Stabilisce un collegamento prima.
Flusso continuo.
Garantisce ordine e arrivo.
- **Esempi:** Posta elettronica, trasferimento file (zero tolleranza per i dati persi).

Non Orientati alla Connessione



- **Caratteristiche:** Messaggi singoli e indipendenti.
Nessun pre-collegamento.
Nessuna garanzia.
- **Esempi:** Streaming, gaming (preferiscono la rapidità alla certezza assoluta).

Attenzione agli Errori Comuni (Checklist Diagnostica)

Errore 1: Porta e Socket sono sinonimi.

Verità: La Porta è solo un dato (un numero). Il Socket è lo strumento (l'oggetto su cui il programma legge e scrive).

Errore 2: Il socket sa cosa sta trasportando.

Verità: Il socket trasporta solo byte crudi. È il Protocollo Applicativo che dà un significato a quei byte (es. HTTP).

Errore 3: Aprire un socket garantisce la connessione.

Verità: Se dall'altra parte non c'è un programma con un socket in ascolto sulla porta giusta, il tentativo fallisce nel vuoto.

Domanda di Autoverifica



L'equazione di rete completa! Anche se il **Socket Remoto** (Server) è identico, i **Socket Locali** (IP/Porta del client) generati per lo Studente A e lo Studente B sono diversi. **La coppia è unica.**

(Sintesi applicativa: La connessione, l'uso e la chiusura saranno esplorati nel prossimo modulo sulle primitive di rete)