



Architetture client-server



Sul libro...

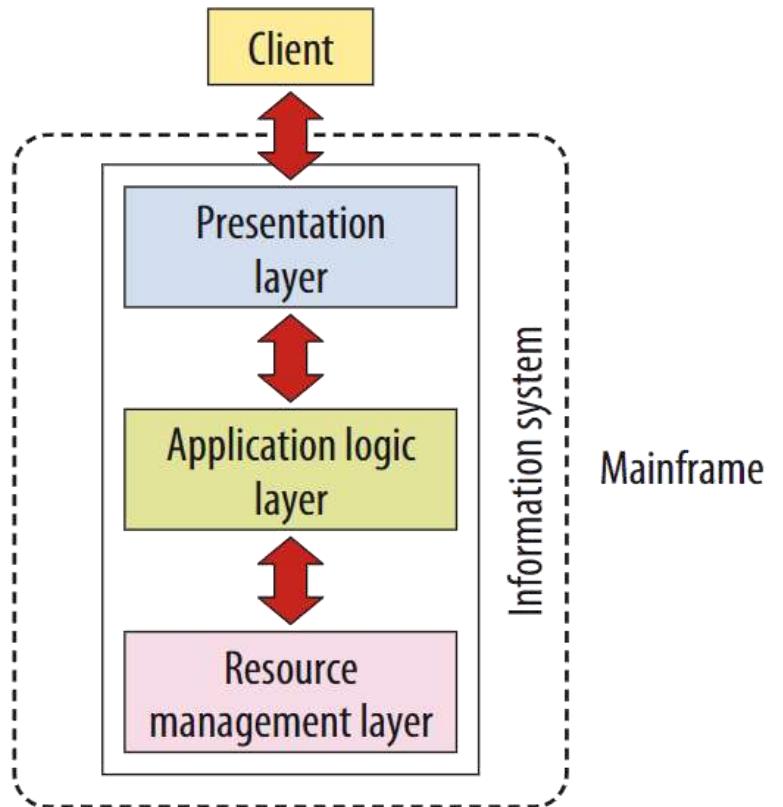
- Unità 1 - Lezione 4
- Da pag. 36 a pag. 42



Architettura a un livello (Single tier)

➤ Anni '60 - '80: mainframe

- Il mainframe è una macchina potente che effettua tutte le operazioni sui dati (memorizzazione, elaborazione e visualizzazione)
- Tutti gli utenti accedono a questo mainframe tramite terminali "stupidi" (uno schermo e una tastiera senza capacità di elaborazione propria)



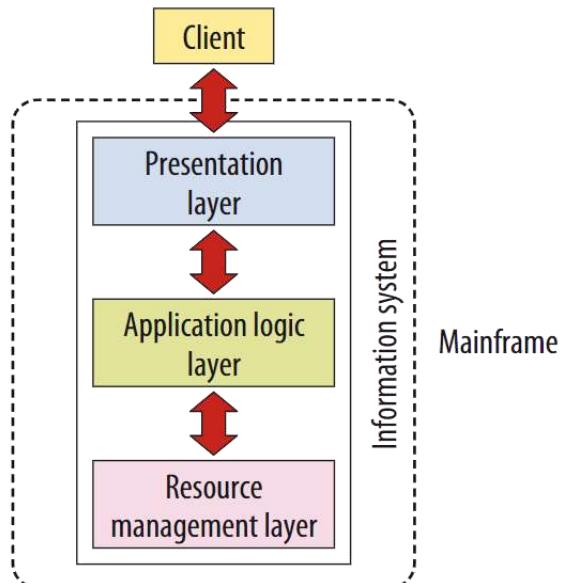
Architettura a un livello (Single tier)



- Gestione centralizzata
(backup e controlli molto semplici)
 - Alta sicurezza



- ▶ Costo elevatissimo
 - ▶ Scarsa scalabilità
 - ▶ Forte dipendenza dal server centrale (se il mainframe si blocca, nessuno lavora)





Modello client server

Client



Il **client** è il software (e per estensione il computer su cui è eseguito) che istanzia l'interfaccia utente di un'applicazione connettendosi tramite un'infrastruttura di rete a un server.

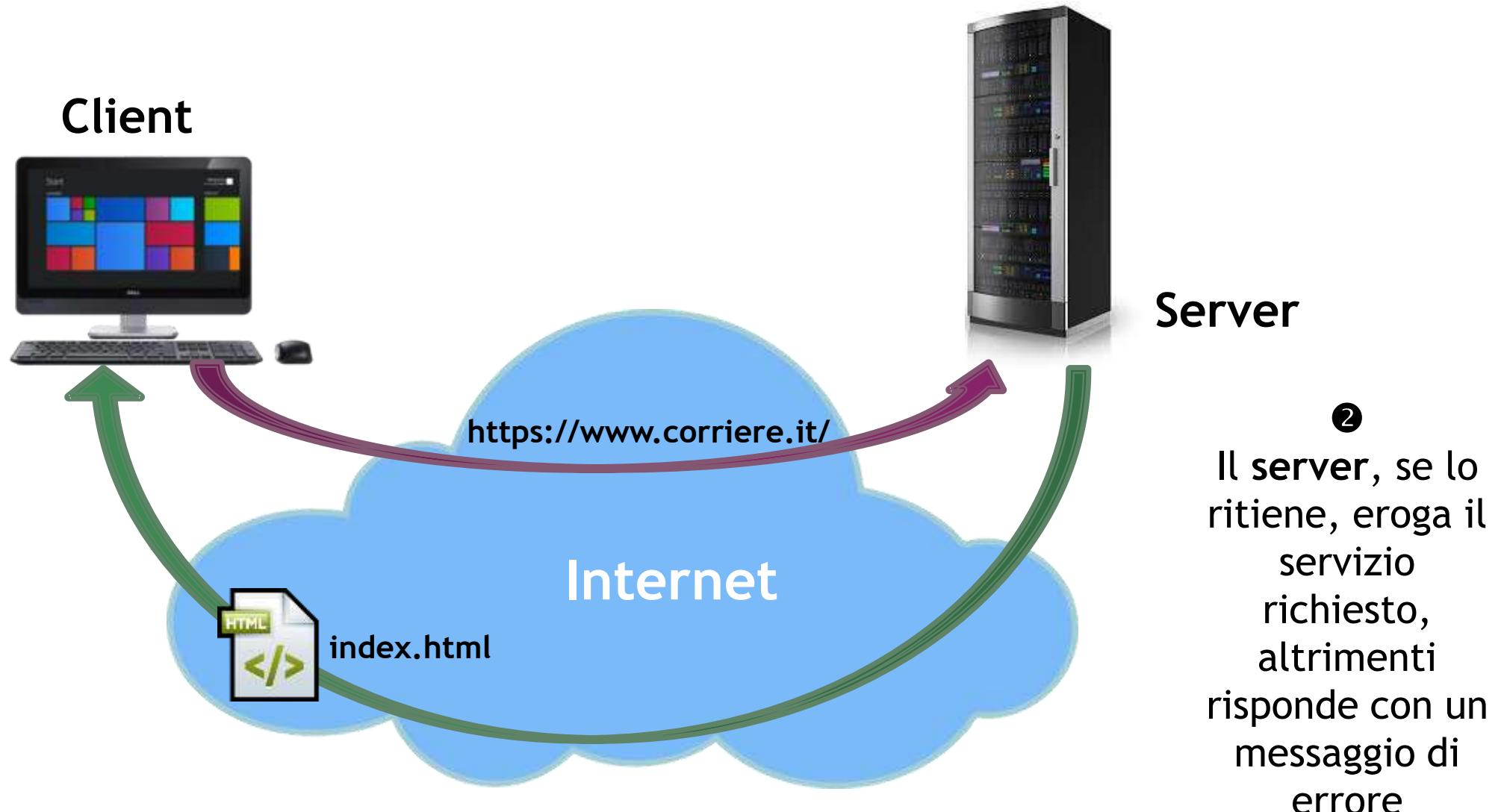
Server



Il **server** è il software (e per estensione il computer su cui è eseguito) che, oltre alla fornitura di servizi ai client, deve implementare le tecniche di gestione degli accessi, allocazione e rilascio delle risorse, condivisione e sicurezza dei dati.

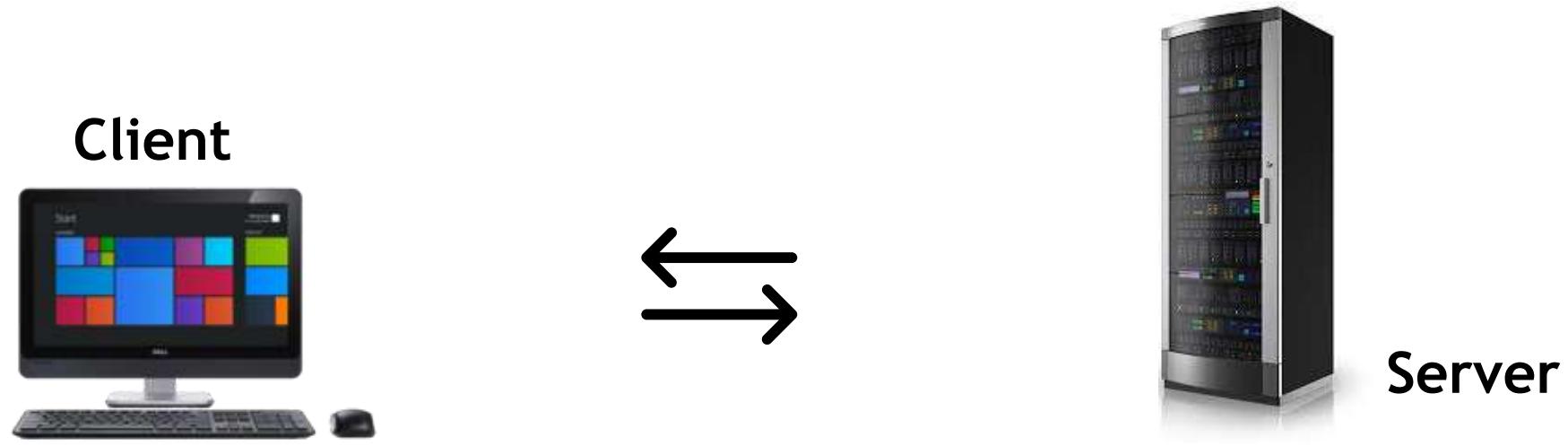


Modello client server





Modello client server



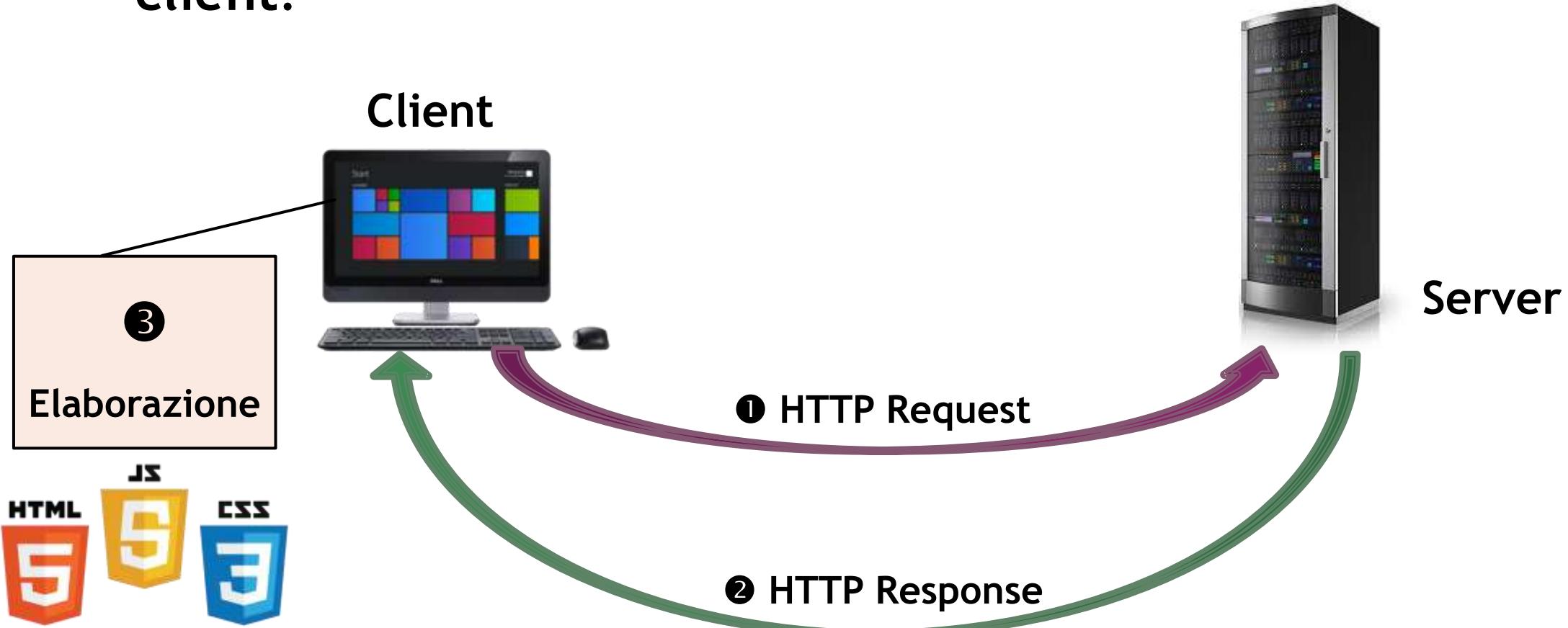
Per interagire tra loro, client e server devono utilizzare un linguaggio comune, ossia un **protocollo**





Tecnologie client-side

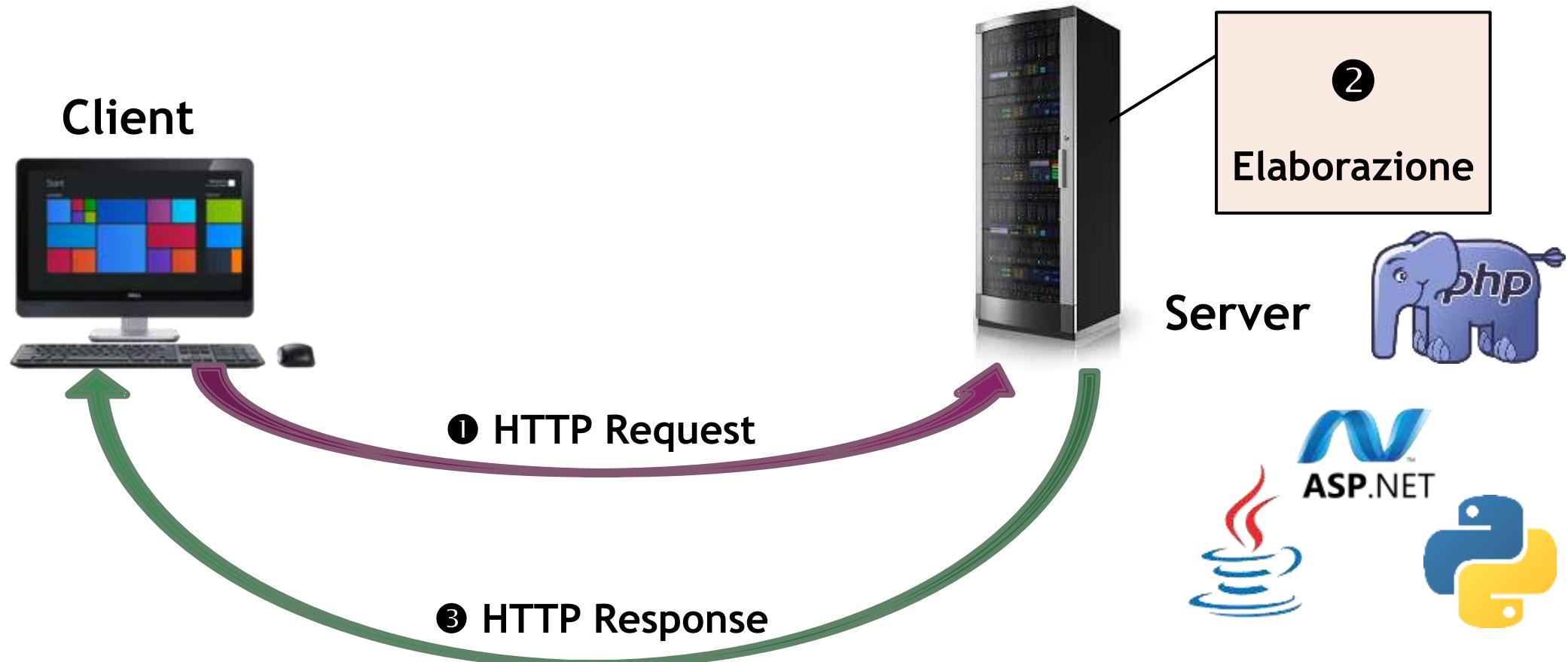
- Sono le strutture tecnologiche, linguaggi di programmazione, librerie e strumenti che vengono utilizzati ed eseguiti sul client.

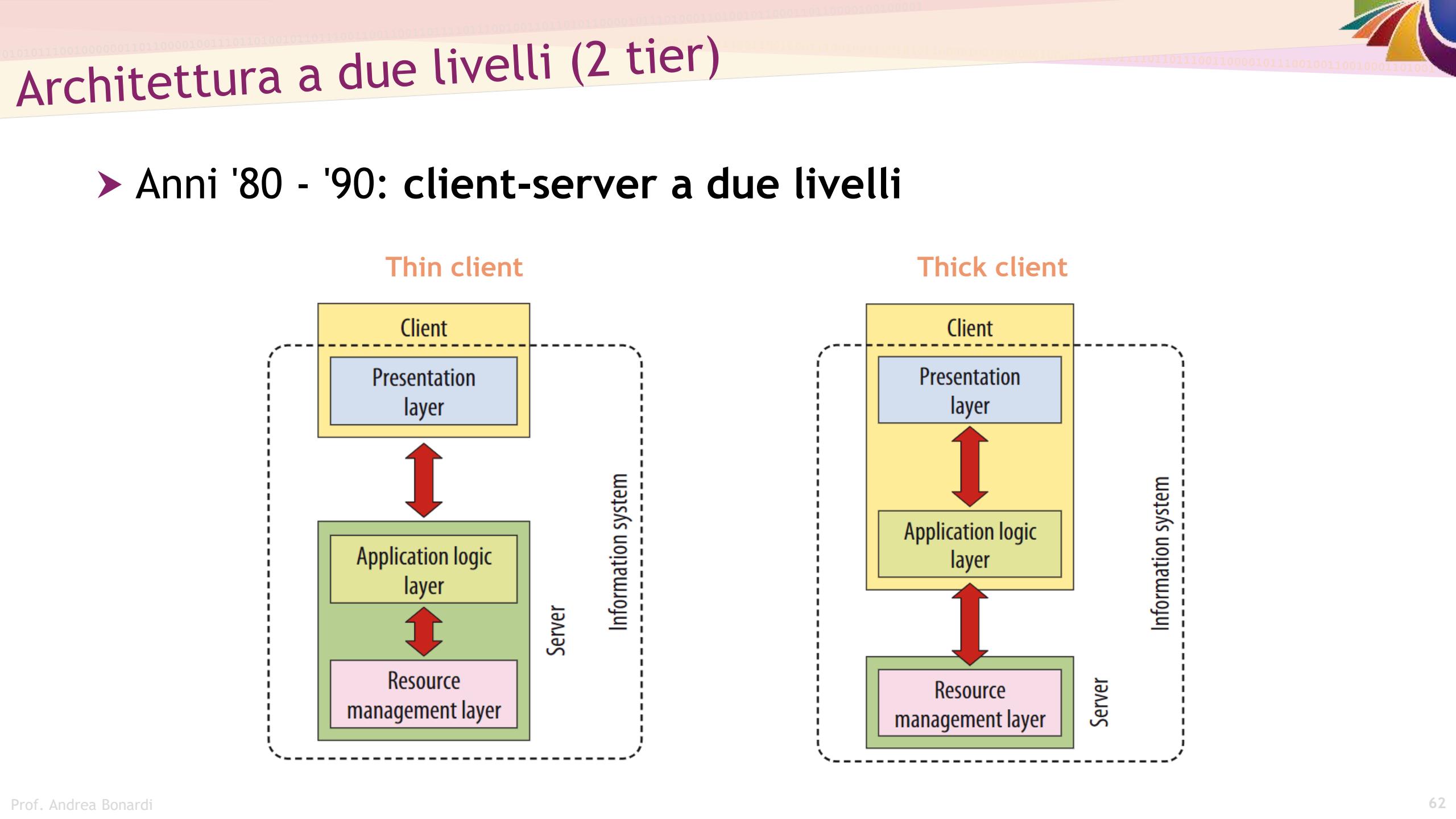




Tecnologie server-side

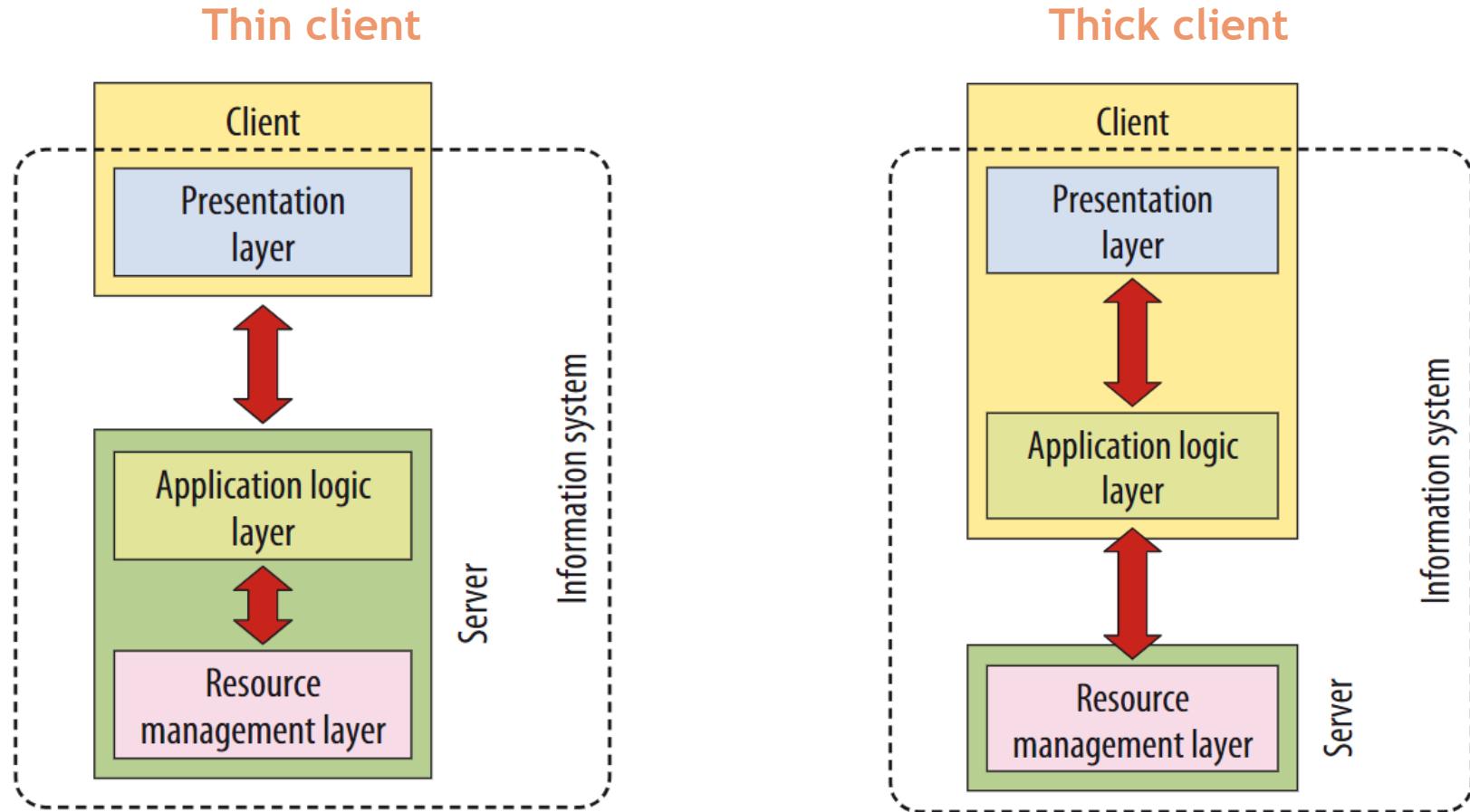
- Sono le strutture tecnologiche, linguaggi di programmazione, framework e risorse che operano sul **server**.





Architettura a due livelli (2 tier)

➤ Anni '80 - '90: client-server a due livelli





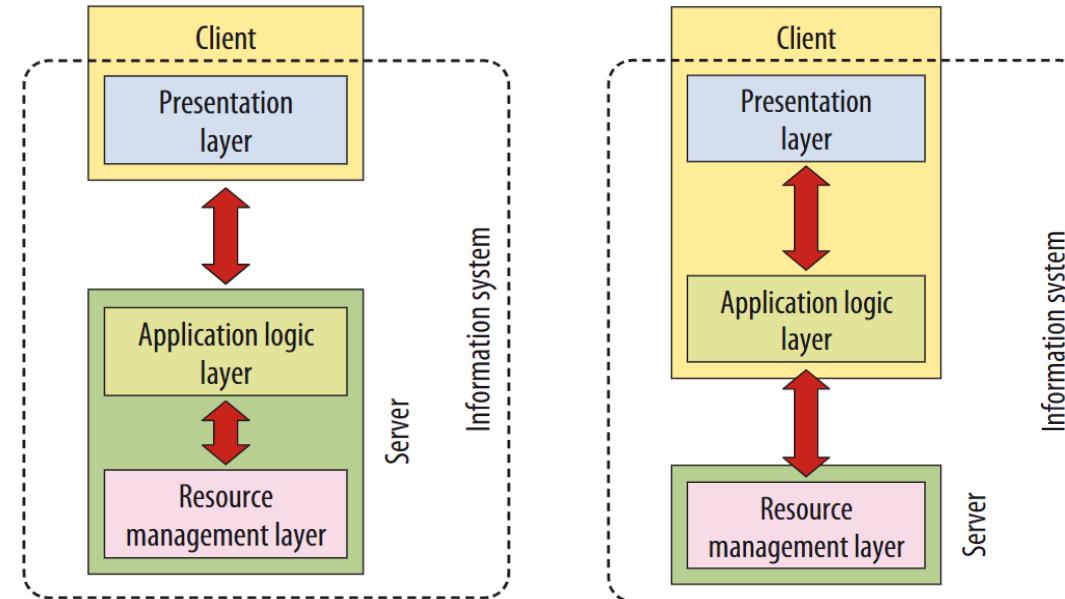
Architettura a due livelli (2 tier)

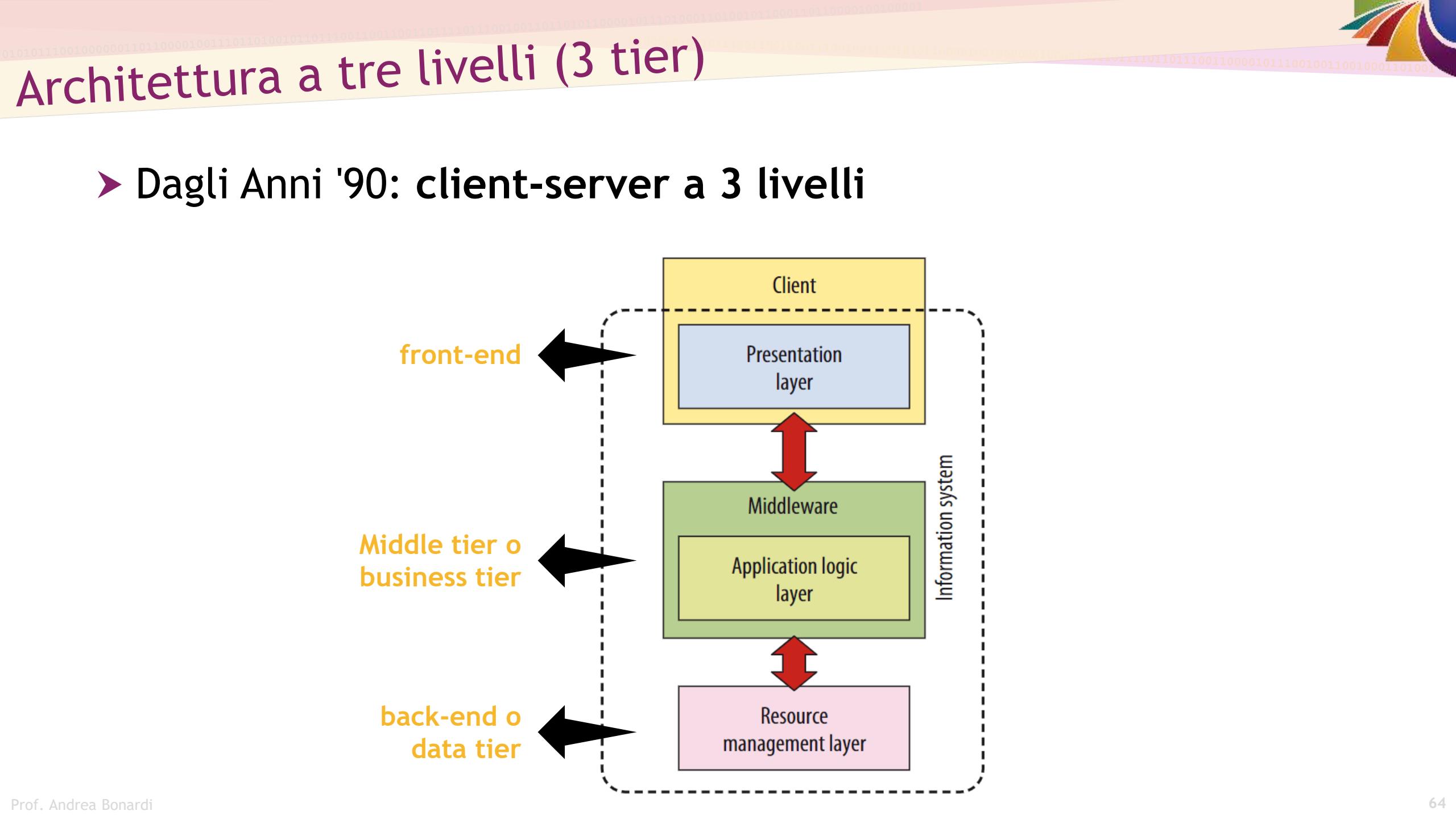


- Migliore distribuzione del carico in quanto sfrutta la potenza di calcolo dei client
- Costi ridotti (viene utilizzato hardware standard)



- Manutenzione complessa (ogni modifica al software richiede aggiornare tutti i client)
- Minore sicurezza







Architettura a tre livelli (3 tier)



- Separazione netta tra interfaccia, logica e dati
- Sicurezza maggiore: Il database è "nascosto" dietro lo strato logico
- Manutenzione e aggiornamenti semplici
- Alta scalabilità



- ▶ Complessità di progettazione, sviluppo e amministrazione
- ▶ Costi maggiori
- ▶ Maggiore overhead di rete

