

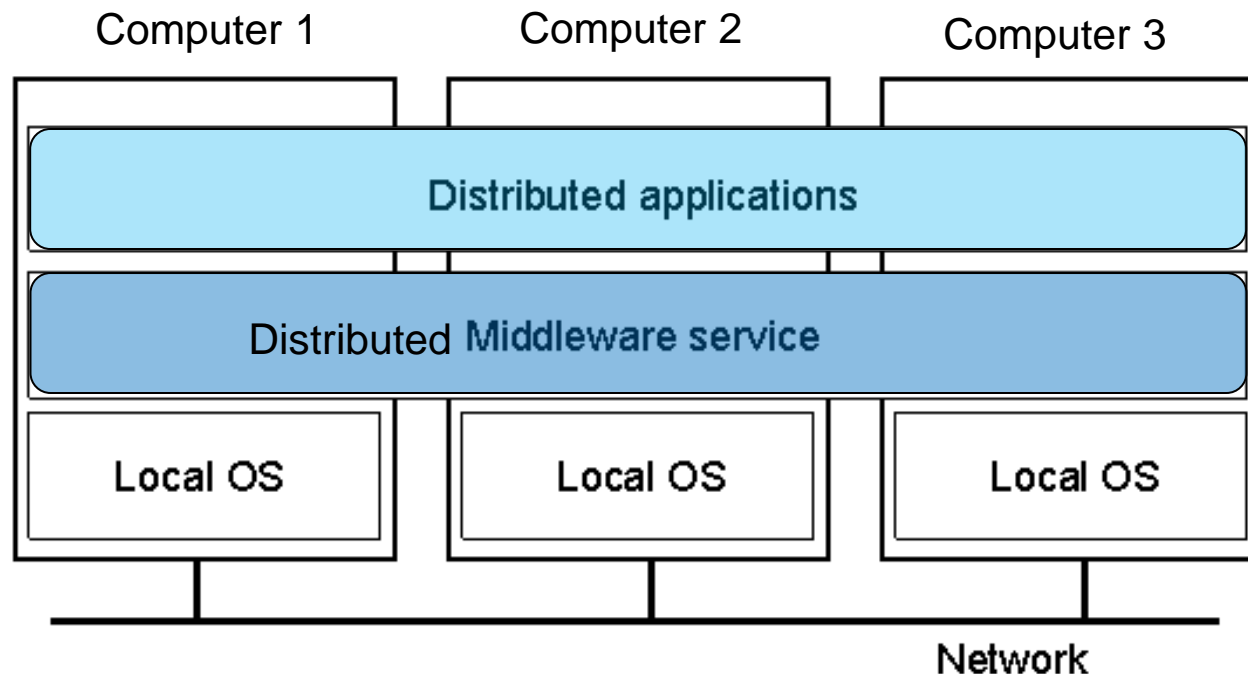
Introduzione ai Sistemi Distribuiti

Definizione di Sistema Distribuito (1)

Un sistema distribuito è:

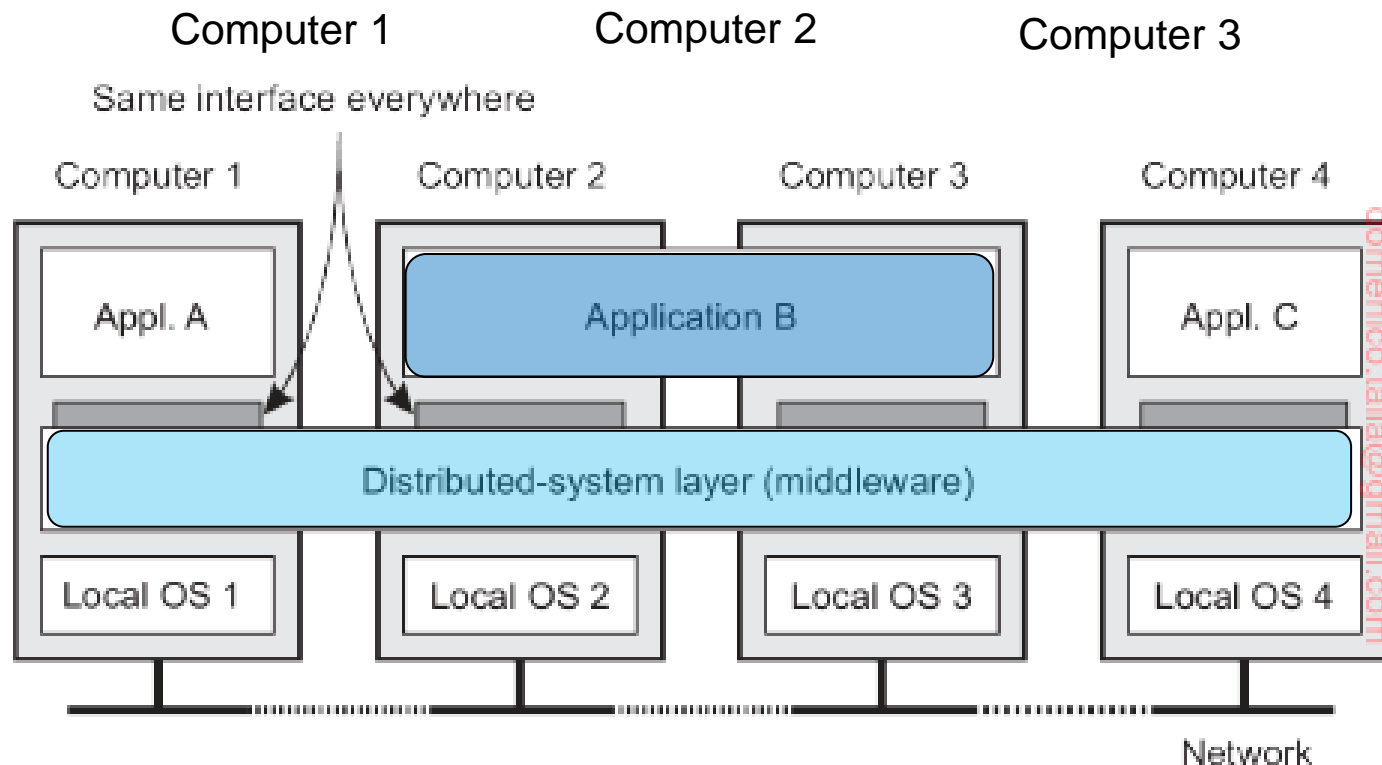
Una collezione di computer *indipendenti*
che appaiono agli utenti come un *sistema*
singolo coerente.

Definizione di Sistema Distribuito (2)



Un sistema distribuito organizzato come middleware.
(Si noti che il Middleware layer si estende su tutte le macchine.)

Definizione di Sistema Distribuito (2)



Un sistema distribuito organizzato come middleware.
(Si noti che il Middleware layer si estende su tutte le macchine.)

Esempi di Sistemi Distribuiti

- Una rete di workstations.
- Una rete Peer-to-Peer.
- Un sistema di prenotazione di voli.
- Il World Wide Web.
- Una Griglia computazionale.
- Un sistema Cloud.
- Una rete di sensori.
- Un insieme di cellulari connessi tra loro.

Obiettivi dei Sistemi Distribuiti

- Connettere utenti e risorse.
- Offrire Trasparenza.
- Apertura al mondo esterno.
- Realizzare sistemi scalabili.
- Aumentare le prestazioni.
- Maggiore tolleranza ai guasti.

Trasparenza in un Sistema Distribuito

Trasparenza	Descrizione
<i>Accesso</i>	Nasconde le differenze nella rappresentazione dei dati e su come una risorsa è acceduta
<i>Locazione</i>	Nasconde dove una risorsa si trovi
<i>Migrazione</i>	Nasconde che una risorsa si possa spostare in una differente locazione
<i>Rilocazione</i>	Nasconde che una risorsa si possa spostare in una differente locazione mentre viene usata
<i>Replicazione</i>	Nasconde che una risorsa possa essere replicata
<i>Concorrenza</i>	Nasconde che una risorsa possa essere condivisa da più utenti concorrenti
<i>Fallimenti</i>	Nasconde il guasto e il recovery di una risorsa (prossima slide)
<i>Persistenza</i>	Nasconde se una risorsa (software) è in memoria o nel disco.

Forme differenti di trasparenza in un sistema distribuito.

Sistema Distribuito

Definizione di Lamport :

“tu sai di averne uno quando il guasto di un computer di cui non hai mai sentito parlare non ti permette di fare il tuo lavoro.”

Scalabilità

- La **scalabilità** è una proprietà importante per i sistemi distribuiti.
- Un sistema è scalabile se all'aumentare delle sue risorse migliorano proporzionalmente le sue prestazioni (*speedup*).
- La scalabilità permette di mantenere le prestazioni costanti se all'aumentare del carico di lavoro si aumentano le risorse (*scaleup*).
- La scalabilità nelle comunicazioni mantiene le latenze pressochè costanti all'aumentare delle distanze tra i nodi.

Problemi di Scalabilità

Concetto	Esempio
<i>Servizi Centralizzati</i>	Un solo server per tutti gli utenti
<i>Dati Centralizzati</i>	Un singolo elenco telefonico on line
<i>Algoritmi Centralizzati</i>	Effettuare il routing basato su informazioni complete

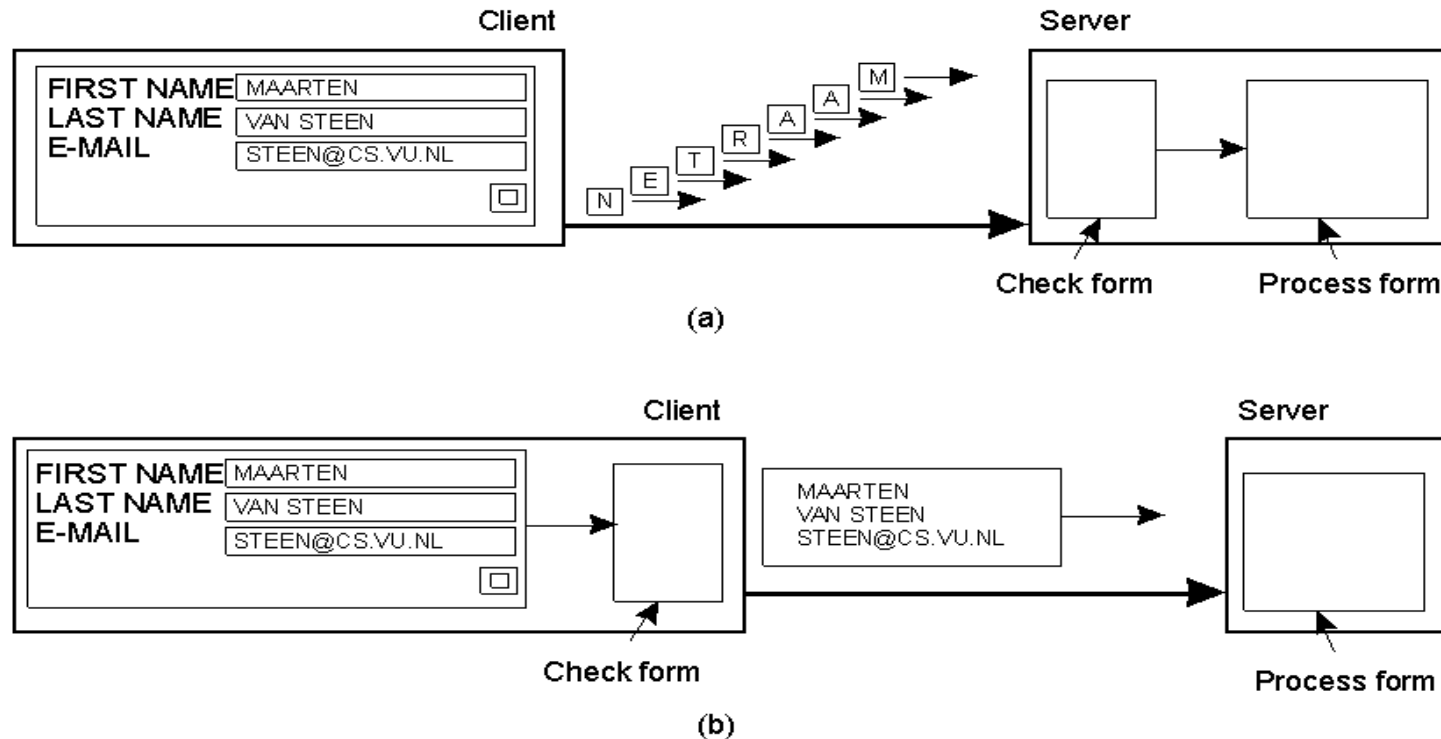
Esempi di **limitazioni** di scalabilità

Algoritmi Decentralizzati

- 1. Nessuna macchina ha l'informazione completa sullo stato del sistema**
2. Le macchine prendono decisioni sulla base di **informazioni locali.**
3. Il fallimento di una macchina non fa fallire l'algoritmo.
4. Non vi è una assunzione di un clock globale.

Più grande è il sistema, maggiore è l'incertezza.

Tecniche di Scalabilità (1)



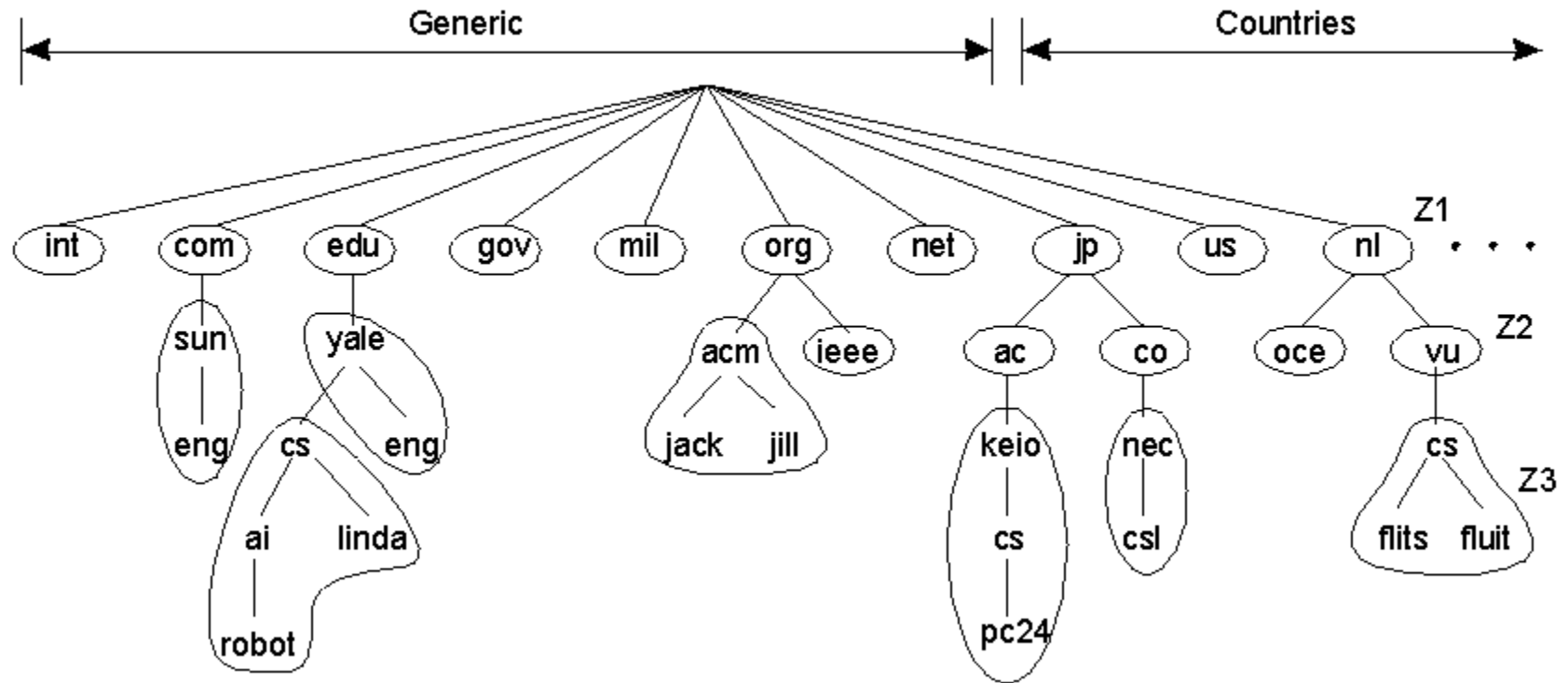
La differenza fra chi tra:

a) un server o

b) un client

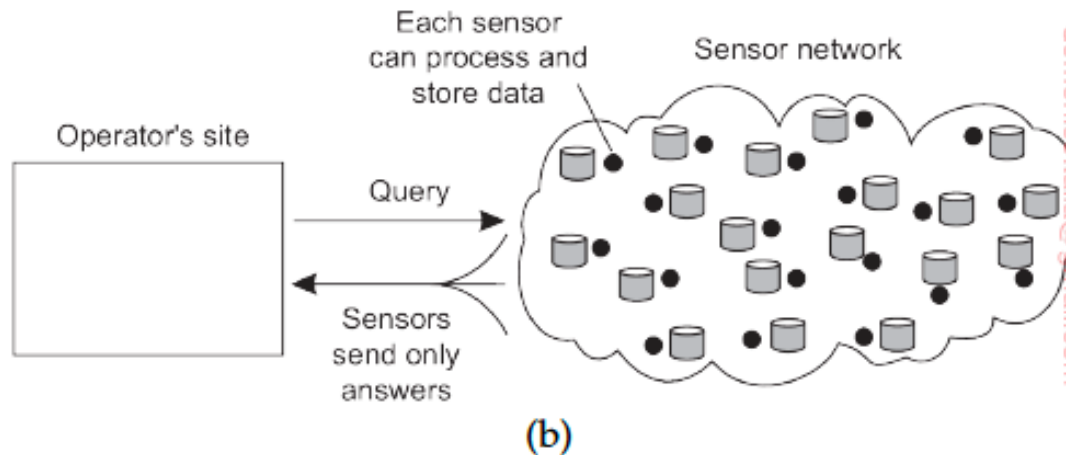
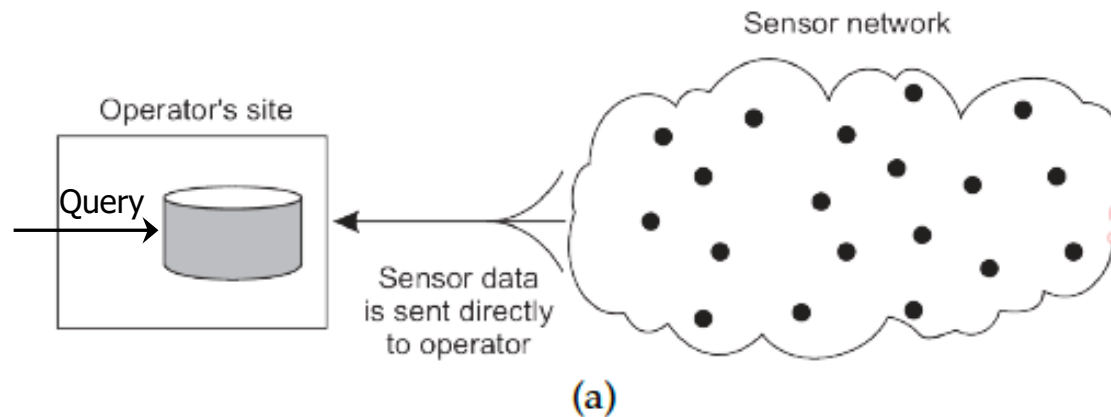
controlla le forme che sono state riempite.

Tecniche di Scalabilità (2)



Un esempio di divisione dello spazio dei nomi del DNS in zone

Tecniche di Scalabilità (3)

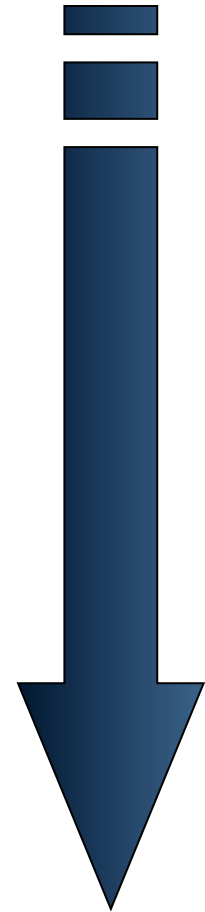


Un database di dati da sensori: soluzioni differenti

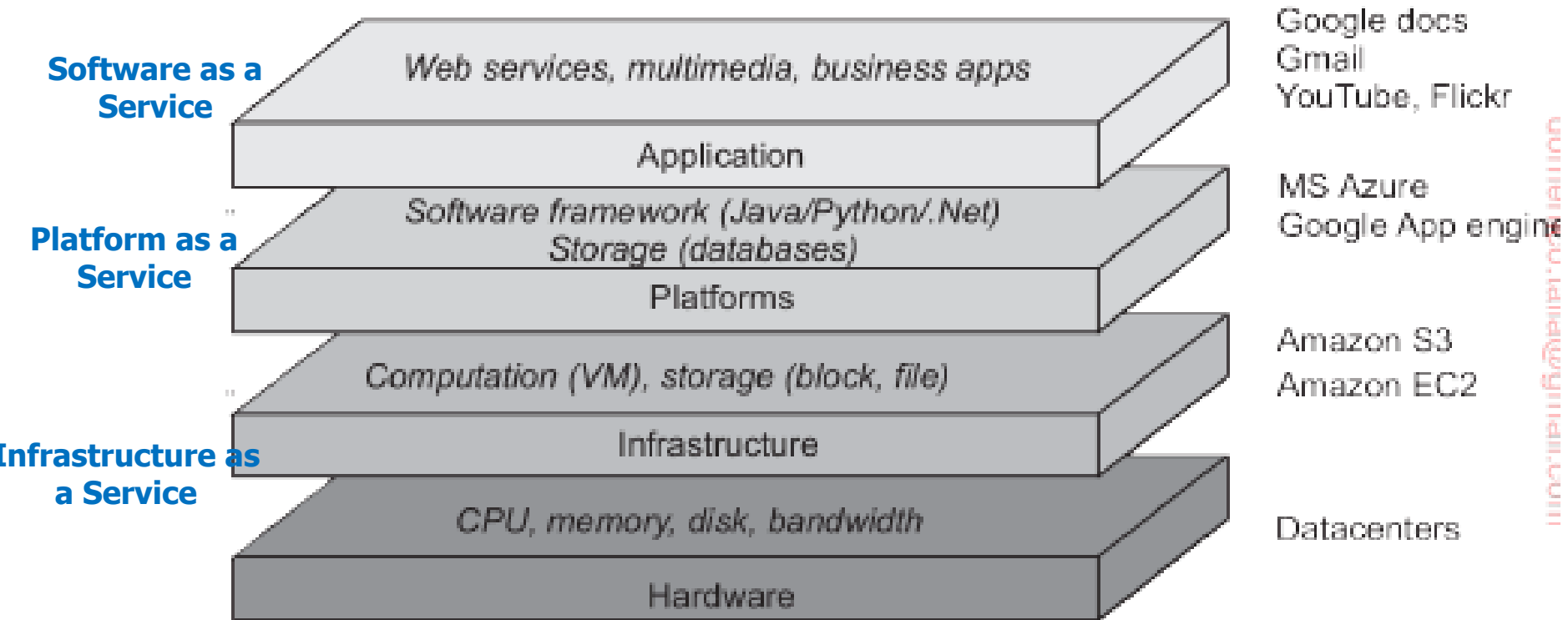
Tipi di Sistemi Distribuiti

- HPC (Multicore/Manycore/Exascale)
- Cluster Computers
- Sistemi Informativi Distribuiti
- Sistemi di Grid e Cloud Computing
- Reti Peer-to-Peer
- Sistemi di Mobile Computing
- Sistemi Ubiqui/Pervasivi (IoT)
- Reti di Sensori/RFID (IoT)

Numero di unità/Granularità

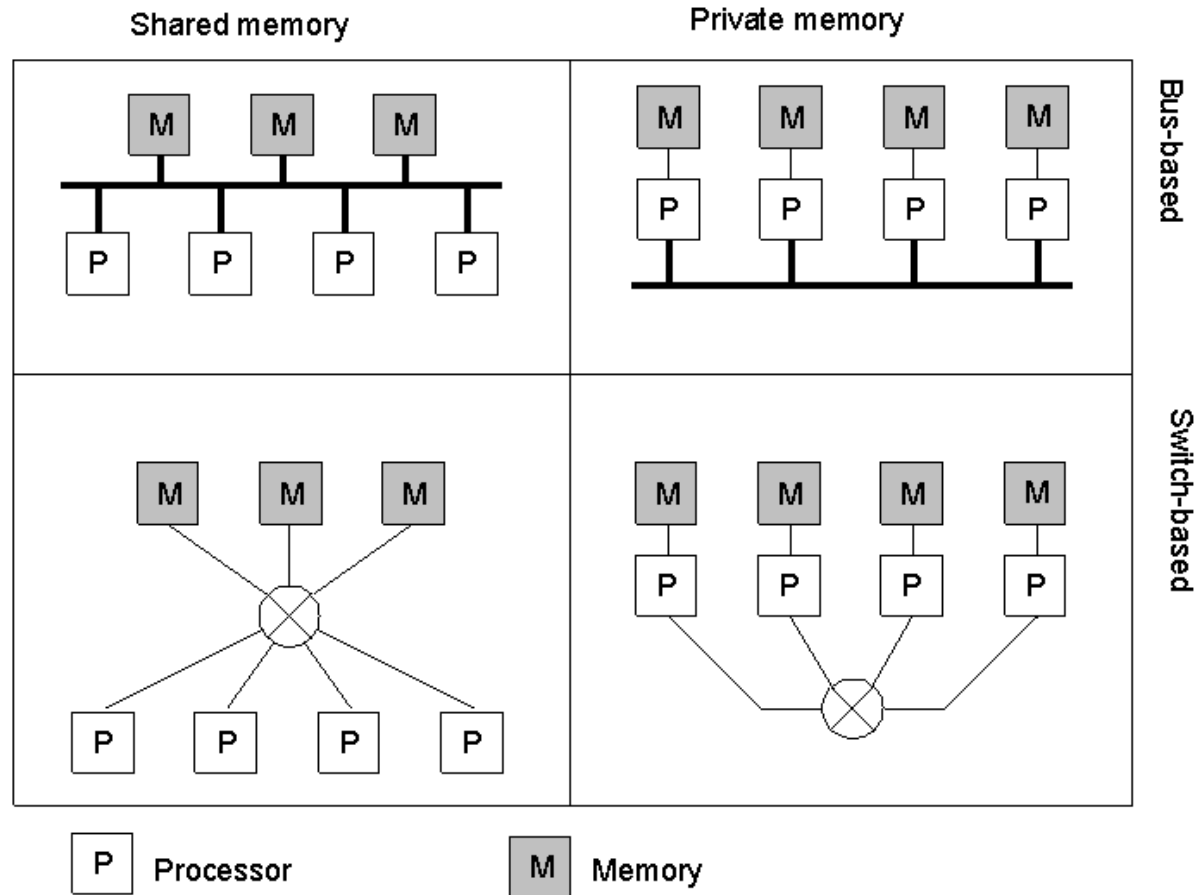


Modelli di Cloud Computing



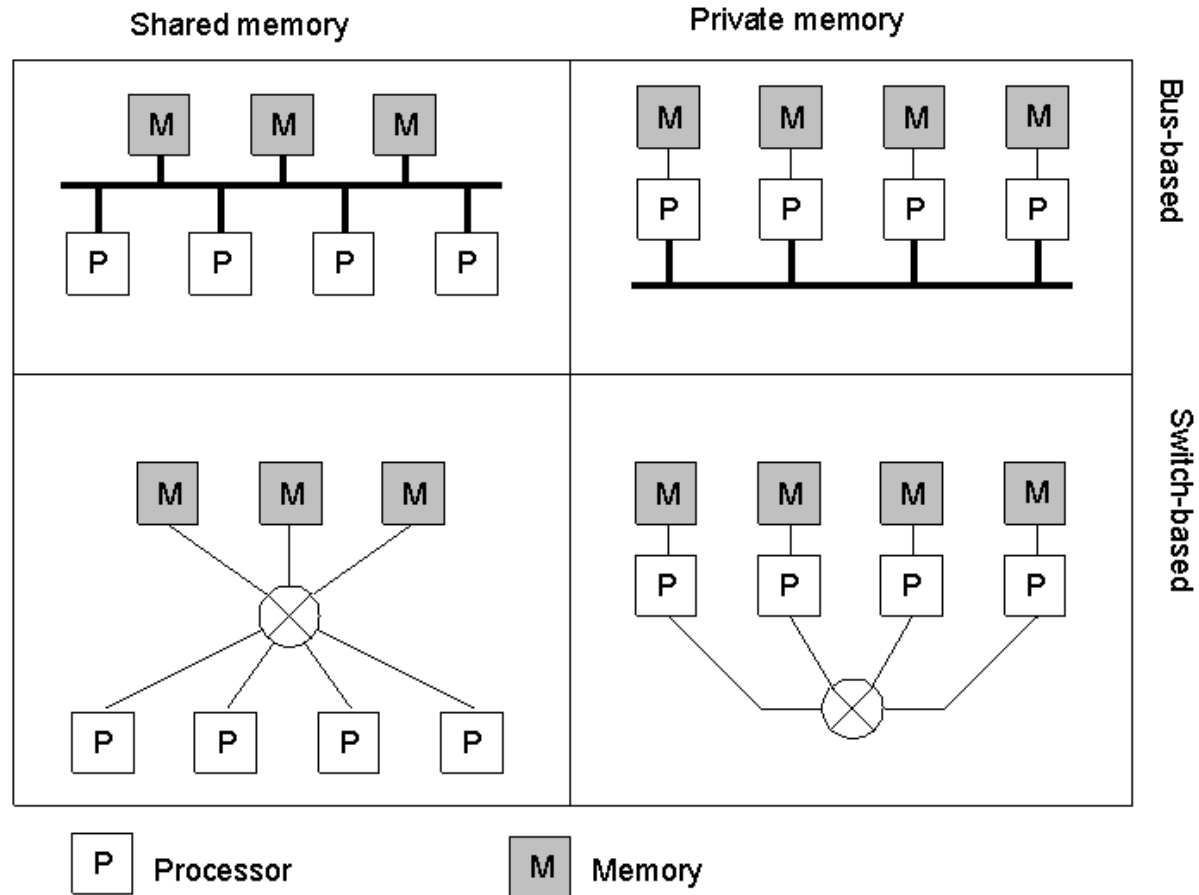
Organizzazione dei principali modelli di Cloud computing

Concetti Hardware



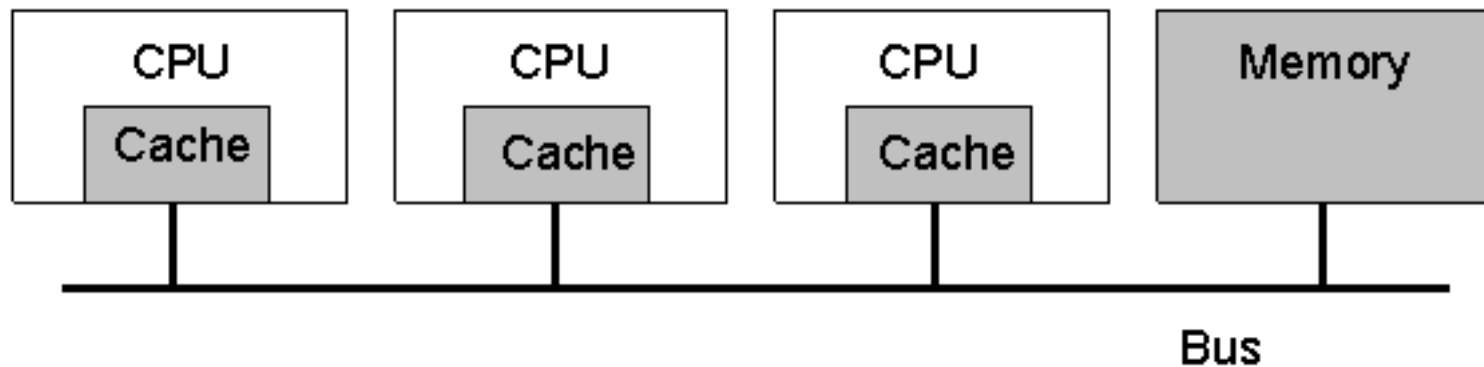
Differenti architetture e memorie in sistemi di elaborazione distribuiti

Concetti Hardware



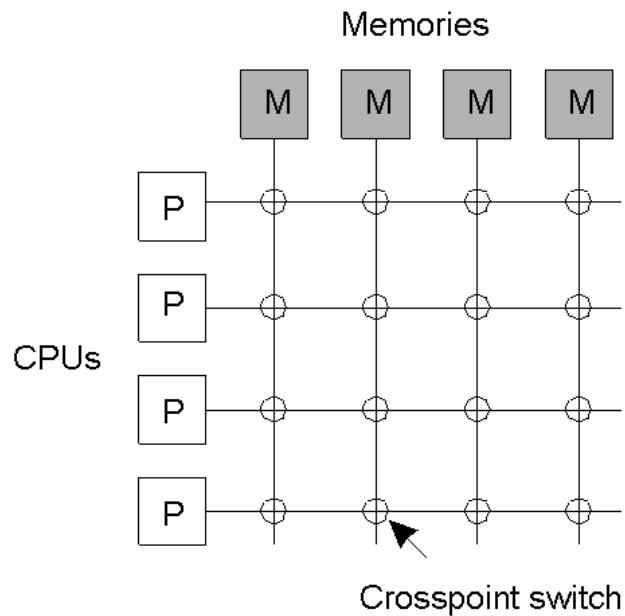
Differenti architetture e memorie in sistemi di elaborazione distribuiti

Multiprocessori (1)

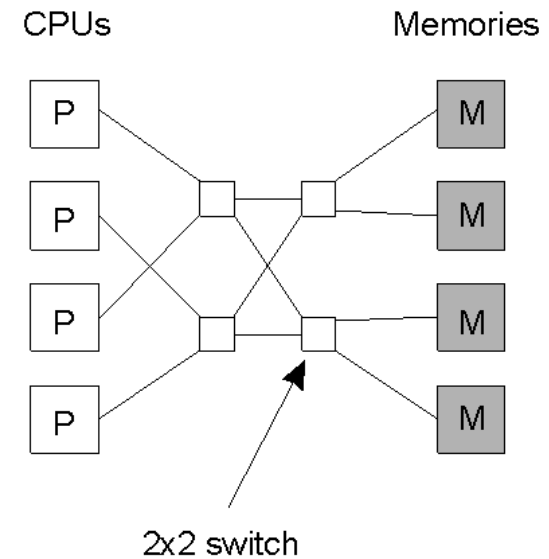


Un multiprocessore basato su architettura a bus.

Multiprocessors (2)



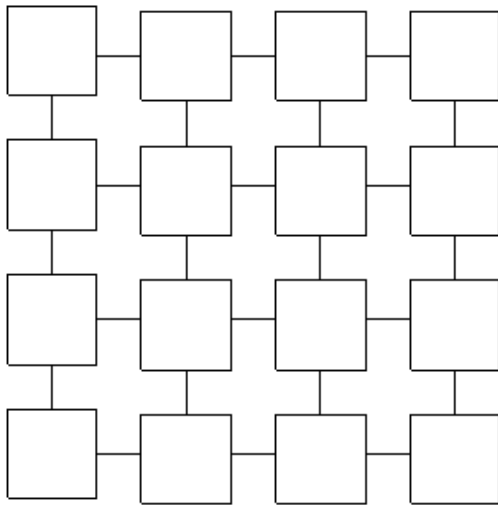
(a)



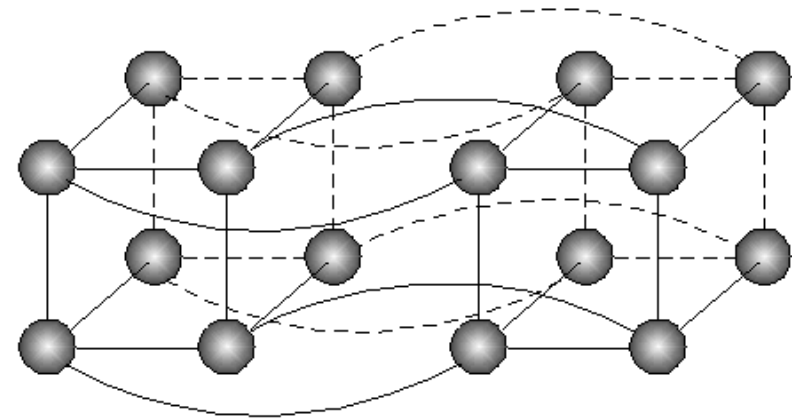
(b)

- a) Un crossbar switch
- b) Una rete omega

Multicomputer Systems Omogenei



(a)

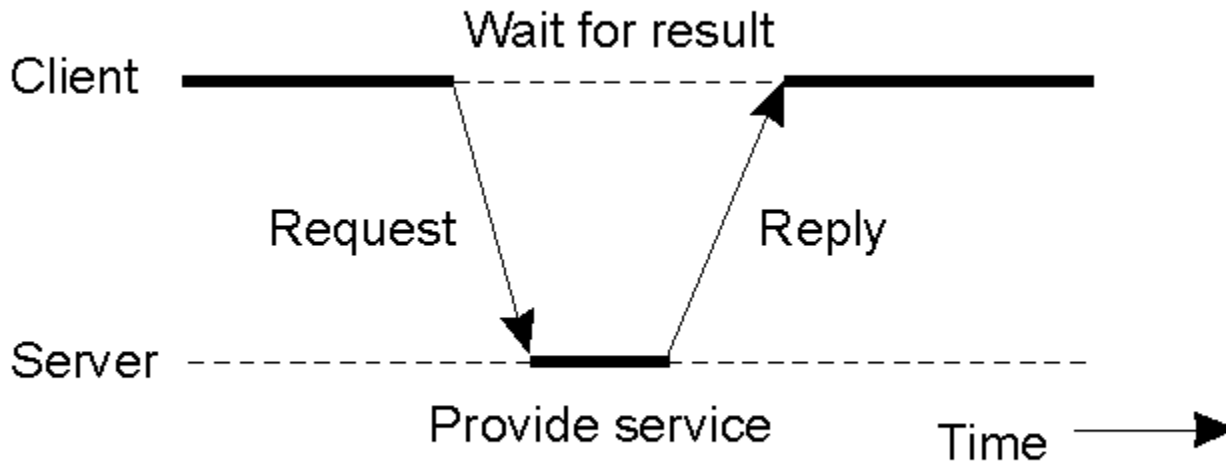


(b)

- a) Mesh
- b) Hypercube

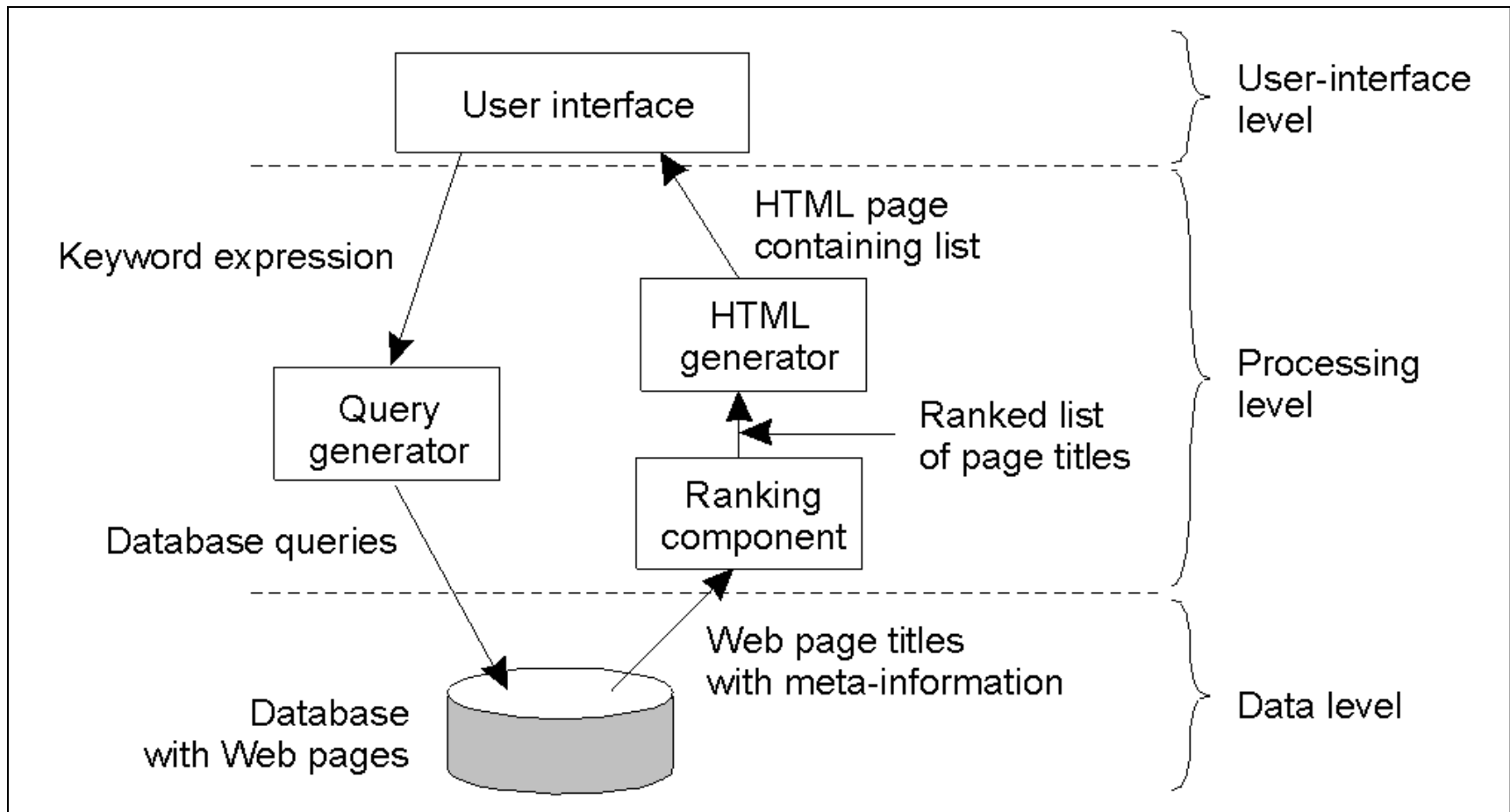
Architetture Server-based

Modello Client-Server



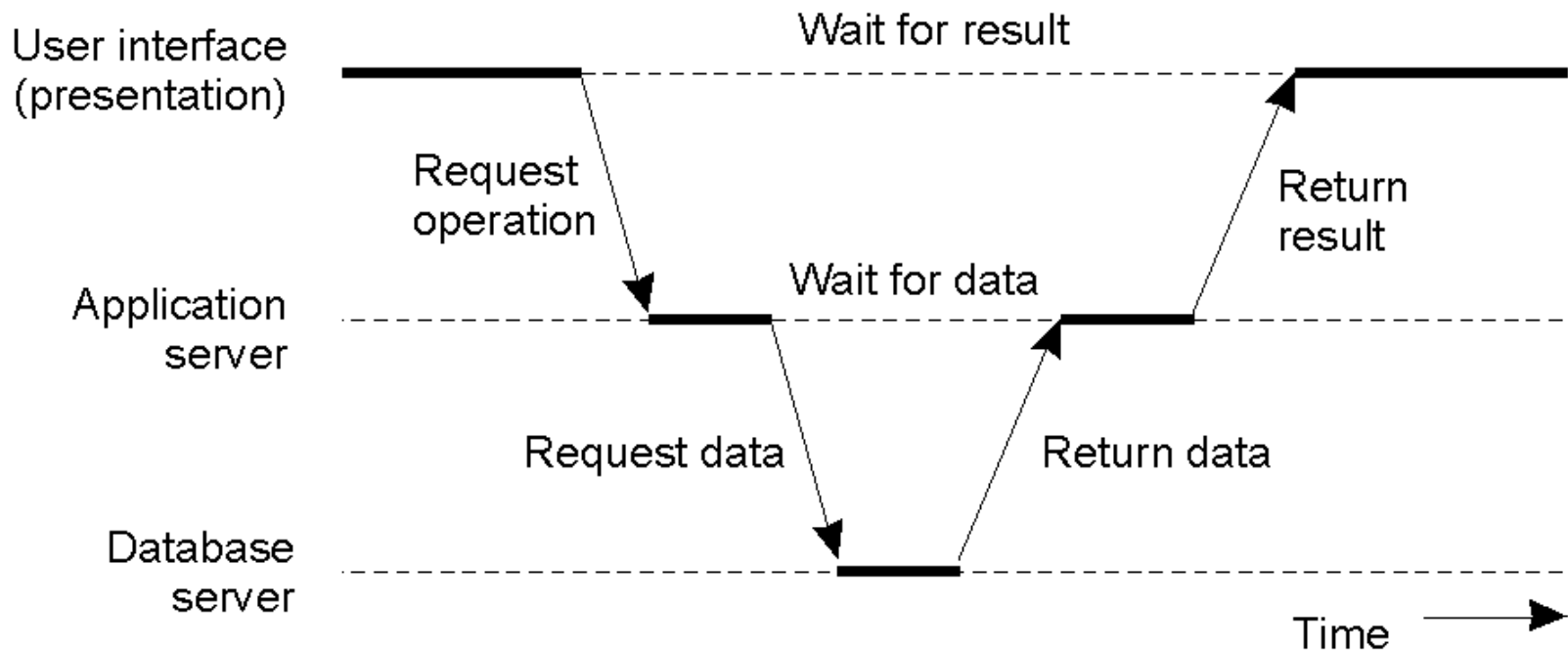
Interazione generale tra un cliente e un server

Livelli di Elaborazione



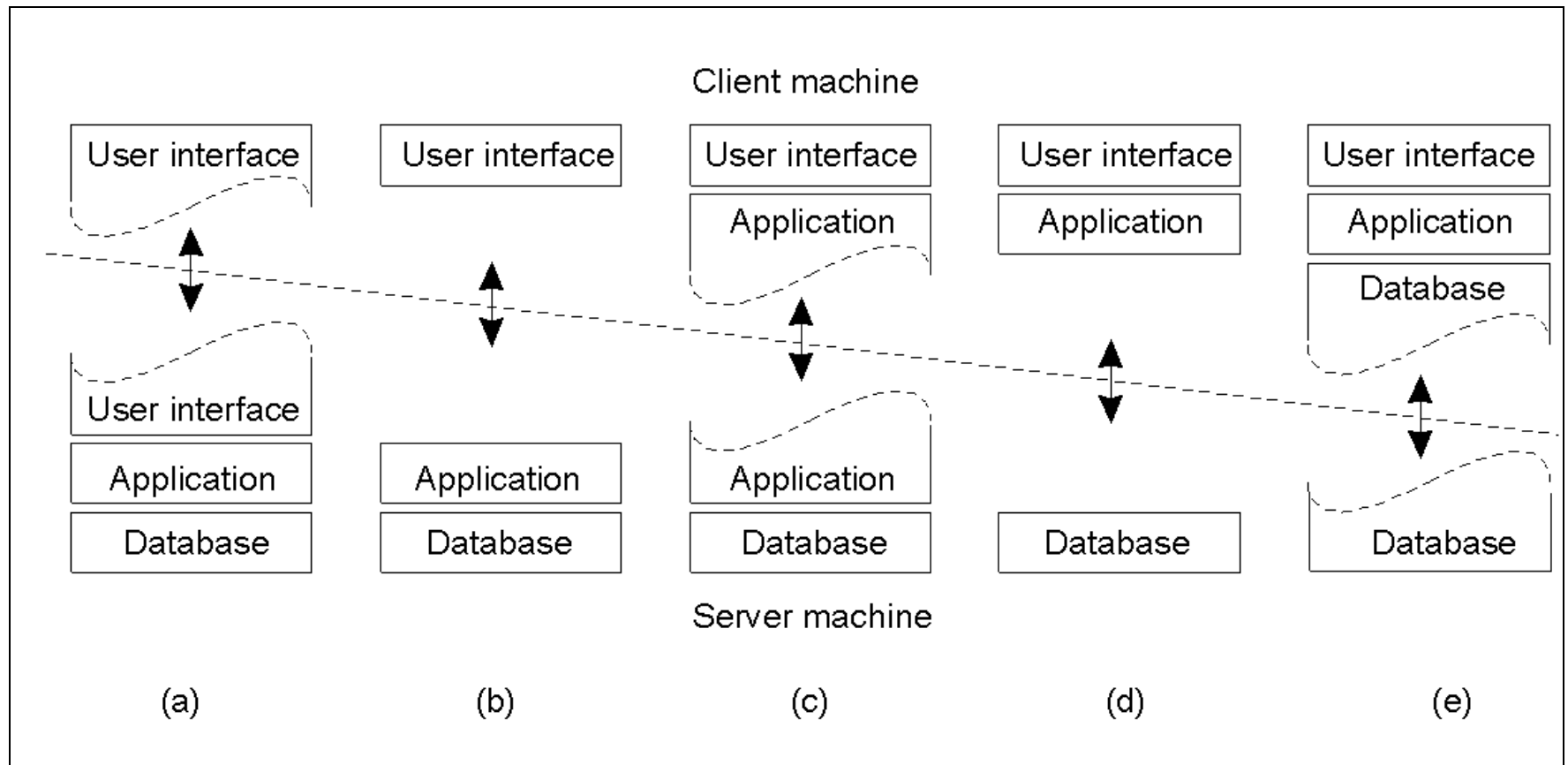
Organizzazione generale di una Internet search engine in tre livelli differenti

Architetture Multi-tiered (1)



Un esempio di un server che opera come un cliente.

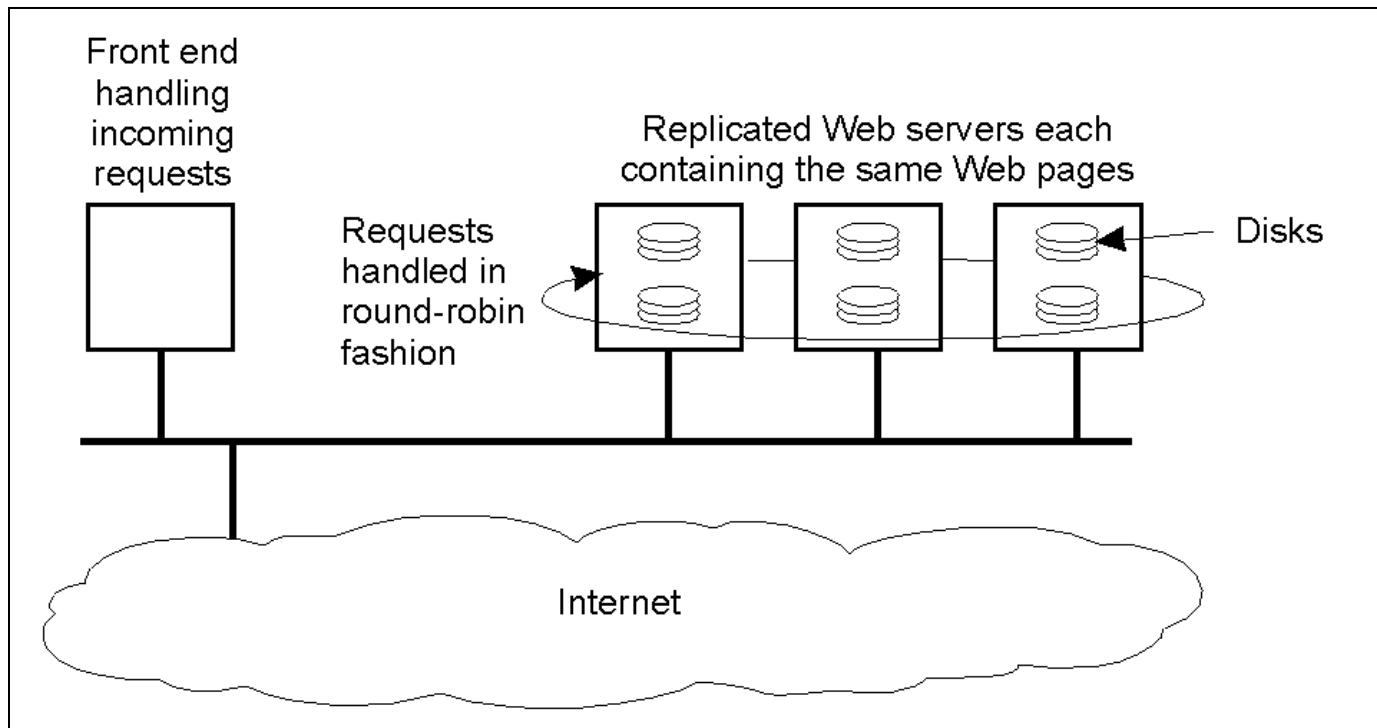
Architettura Multi-tiered (2)



Organizzazioni client-server alternative (a) – (e).

Architetture Decentralizzate

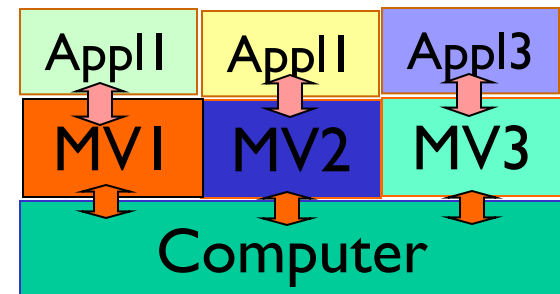
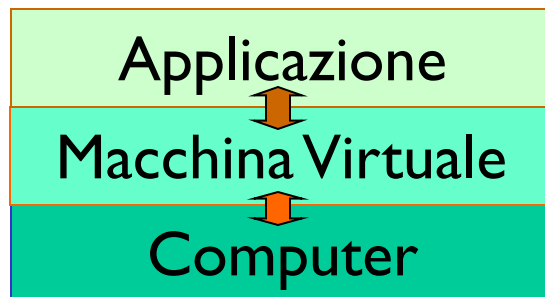
Sistemi multi-server, sistemi peer-to-peer



Un esempio di distribuzione orizzontale di un servizio Web

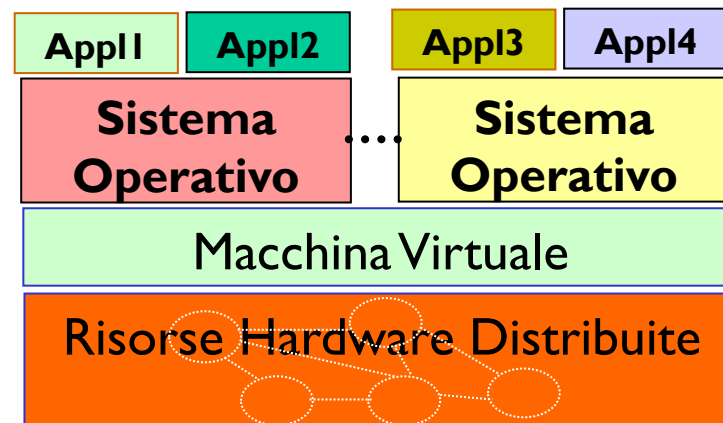
Macchine Virtuali

- Il concetto di Macchina Virtuale è stato definito molto tempo fa (anni ' 60) in un contesto centralizzato.
- Una Macchina Virtuale permette di rappresentare le risorse hardware diversamente dai loro limiti fisici. Un singolo computer puo' essere rappresentato e usato come differenti ambienti di elaborazione.



Macchine Virtuali in Sistemi Distribuiti

- In un sistema distribuito tramite le Macchine Virtuali si possono virtualizzare risorse remote per comporre sistemi distribuiti virtuali che integrano risorse hardware (RAM, CPU, ...) presenti in siti remoti.
- Un livello di Macchina Virtuale nasconde la diversità delle piattaforme ed offre un ambiente omogeneo e “confezionato” sulle esigenze degli utenti e delle loro applicazioni



Macchine Virtuali nei Sistemi Distribuiti

- Secondo questo approccio si può “costruire” dinamicamente un sistema distribuito composto da diverse CPU, memorie, dischi, reti, device, etc.
- L’approccio basato su una macchina virtuale permette di sviluppare applicazioni distribuite che usino molte macchine in maniera trasparente. (ES: JVM ?).

