

Classificazione dei sistemi di elaborazione

- Funzione e interazione con l'utente
 - Batch
 - Interattivi
 - Dedicati
 - In tempo reale
- Caratteristiche hardware
 - Distribuiti
 - Paralleli
 - Mobili
 - Embedded
- Modalità di gestione del tempo di CPU
 - Mono-programmati
 - Multi-programmati
 - A divisione di tempo (time-sharing)

Sistemi di elaborazione: Un po' di storia

- Prima generazione 1945 - 1955
 - valvole, schede perforate
- Seconda generazione 1955 - 1965
 - transistor, sistemi batch
- Terza generazione 1965 – 1980
 - Circuiti integrati, sistemi multi programmati
- Quarta generazione 1980 – oggi
 - personal computer

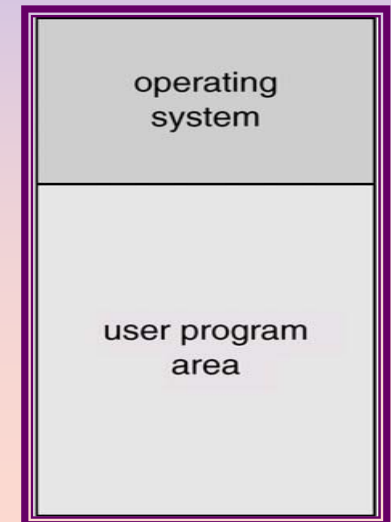
Sistemi Mainframe (Batch)

■ Caratteristiche:

- Sistemi di elaborazione con grandi capacità di calcolo, da condividere per l'esecuzione di numerosi programmi (job)
- programmi batch (non interattivi), eseguiti in sequenza
- Esigenza principale: ridurre il tempo di trasferimento del controllo tra un programma e l'altro.

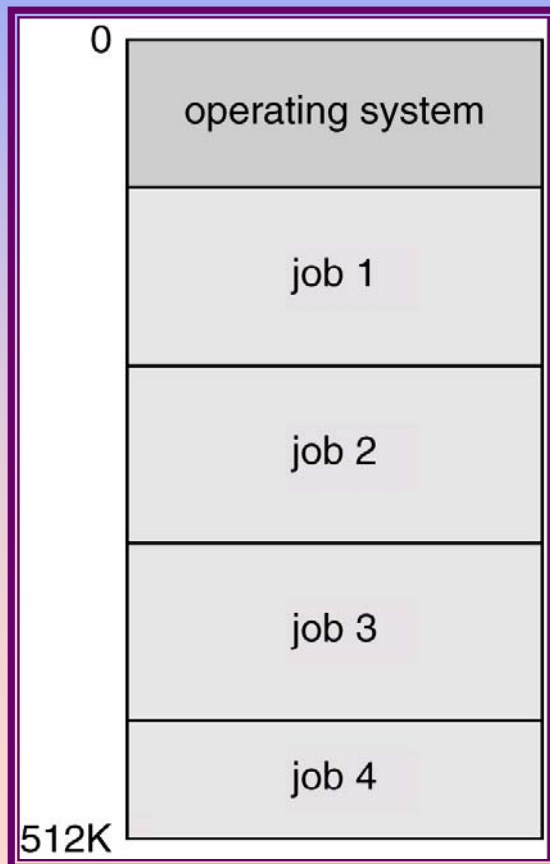
■ Monitor residente:

- primo sistema operativo rudimentale
- I monitor controlla il trasferimento dei job
 - controllo iniziale
 - quando un job è completato il controllo passa al monitor



Sistema Batch Multiprogrammato

Molti programmi (job pool) sono tenuti insieme in memoria e la CPU è assegnata a loro di volta in volta.



Caratteristiche richieste al SO

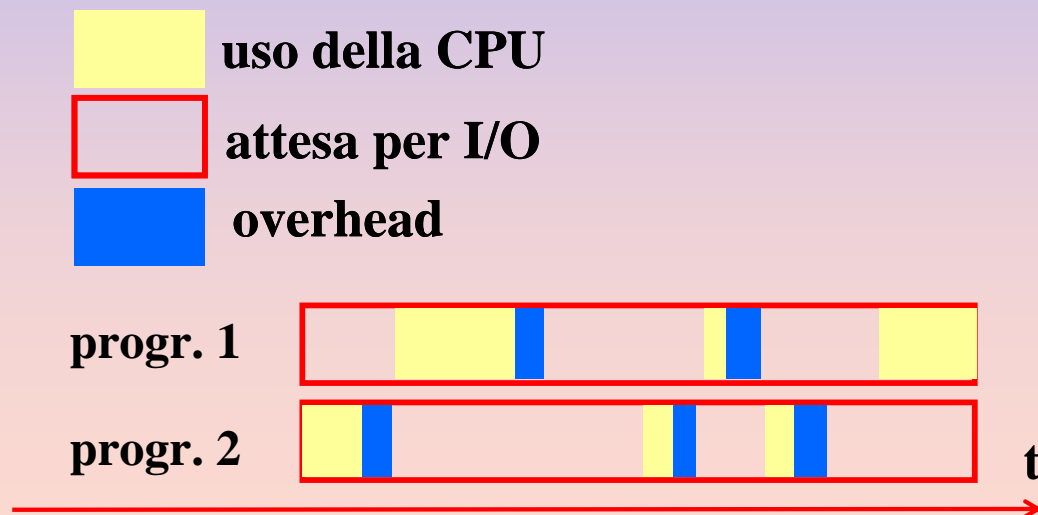
- Gestione della memoria:
 - il SO deve allocare la memoria a più job.
- CPU scheduling:
 - il SO deve scegliere tra più job pronti ad essere eseguiti
- Allocazione dei dispositivi ai job
 - controllo diretto delle operazioni di I/Ox

Sistemi a partizione di tempo (time-sharing)

- Evoluzione della multi-programmazione
- Suddivisione del tempo della CPU
 - La CPU è una risorsa e viene assegnata di volta in volta a diversi programmi tenuti in memoria (processi)
 - Un processo può essere interrotto momentaneamente per far avanzare anche l'esecuzione di altri processi, e poi essere ripreso successivamente.
- Vantaggio: maggiore grado di interattività
 - un file system on-line deve essere disponibile agli utenti per accedere dati e codice.

Sistemi a partizione di tempo (2)

- Ad ogni programma il S.O. assegna ciclicamente un intervallo (quantum) di tempo della CPU, fino al suo completamento
- Al termine dell'intervallo (o durante, se il programma inizia un'operazione di I/O) la CPU viene assegnata ad un altro programma



Sistemi Desktop - PC

- *Personal computer.*

- computer dedicato ad un singolo utente.
- I/O devices – tastiera, mouse, display, piccola stampante.

- Vantaggi

- Convenienza e responsabilità.
- sfruttano tecnologia sviluppata per altri sistemi di elaborazione più complessi

- Può essere dotato di differenti tipi di sistemi operativi (Windows, MacOS, UNIX, Linux).

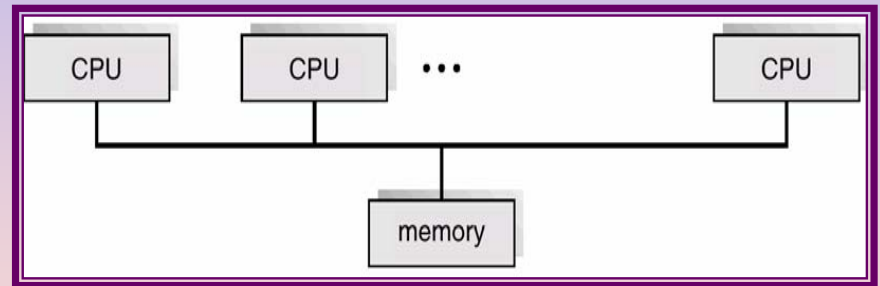
Sistemi Paralleli

■ Caratteristiche

- Multiprocessore:
 - sistema con più di una CPU e con connessioni brevi e veloci.
- Sistemi strettamente accoppiati (*Tightly coupled systems*)
 - i processori condividono la memoria e il clock
 - la comunicazione avviene tramite la memoria condivisa.

■ Vantaggi:

- Prestazioni superiori
- Economicità
- Affidabilità
 - Graceful degradation
 - Tolleranza ai guasti



Sistemi Distribuiti

■ Caratteristiche

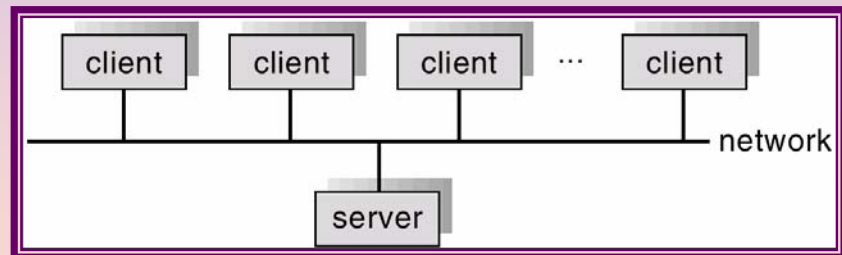
- L'elaborazione è distribuita tra diversi calcolatori collegati tra loro.
- Sistemi debolmente accoppiati (*Loosely coupled systems*)
 - Ogni processore ha la propria memoria locale
 - La comunicazione avviene tramite una rete
 - Non esiste memoria condivisa.

■ Vantaggi

- Condivisione di risorse
- Accelerazione dell'elaborazione
- Affidabilità
- Comunicazione e distribuzione

Sistemi Distribuiti

- Richiedono una infrastruttura di rete:
 - Local area networks (LAN) or Wide area networks (WAN)
- Alcuni sistemi distribuiti:
 - Griglie Computazionali (Grid computing)
 - Applicazioni Web (Web Computing)
- Paradigmi tipici di interazione
 - client-server
 - peer-to-peer



Cluster computer

■ Caratteristiche

- Un cluster computer è composto da un insieme di macchine convenzionali connesse in rete per costituire un singolo sistema con condivisione dei dischi (**Beowulf**)

■ Vantaggi

- Alta affidabilità.
- Basso costo.

■ Sistemi Operativi:

- sistemi operativi ad hoc o sistemi operativi convenzionali con software di supporto per la comunicazione e condivisione dei dati

Sistemi Real-Time

■ Caratteristiche

- Sistemi con vincoli temporali ben definiti sull'elaborazione e sull'accesso alle risorse.
- Sono spesso usati per controllare dispositivi in applicazioni dedicate
 - gestione di auto o di robot, telemedicina, sistemi di controllo militari, ecc.

■ Sistemi Operativi

- Enfasi sull'alta priorità dei processi
- Adottano algoritmi di scheduling specifici

Sistemi mobili

■ Varie tipologie

- Personal Digital Assistants (PDAs), Palmtop, Pocket PC
- E-books
- Telefoni Cellulari (smart-phones)

■ Problemi:

- Memoria limitata
- Processori lenti
- Display piccoli.

Ambienti di elaborazione

- Elaborazione tradizionale
- Web-Based Computing
- Sistemi Embedded

ARCHITETTURA DEI SISTEMI OPERATIVI

Architettura dei Sistemi Operativi

- Servizi di un sistema operativo
- Componenti
- System Call
- Programmi di sistema
- Struttura del sistema operativo
- Macchine virtuali

Servizi di un sistema operativo

- **Esecuzione di programmi** – capacità di caricare in memoria un programma ed eseguirlo (allocazione di memoria e CPU)
- **Operazioni di I/O** – poiché i programmi non possono eseguire direttamente le operazioni di I/O, il sistema operativo deve fornire meccanismi per le operazioni di I/O.
- **Gestione del File system** – fornire ai programmi modalità per leggere, scrivere, creare e cancellare file.
- **Comunicazione** – scambio di informazioni tra processi in esecuzione sullo stesso computer o su computer connessi in rete.
- **Scoperta di errori** – rilevamento di errori nella CPU, nella memoria, nell'I/O o nei programmi utente.

Servizi aggiuzionali

Funzioni del sistema operativo che servono non tanto per l'esecuzione dei programmi utente, ma per assicurare una gestione efficiente delle operazioni del sistema:

- **Allocazione di risorse** – allocazione di risorse per i programmi eseguiti contemporaneamente.
- **Accounting** – memorizzazione dell'uso delle risorse da parte dei vari utenti per il calcolo di costi e per la generazione di statistiche.
- **Protezione** – controllo degli accessi a tutte le risorse del sistema.

Componenti del sistema

Interprete dei comandi
Sistema di protezione
Accesso alla Rete
Gestione della memoria secondaria
Gestione dell'I/O
Gestione dei file
Gestione della memoria principale
Gestione dei processi (nucleo)

Gestione dei Processi

- Un *processo* è un programma in esecuzione.
- Un processo ha bisogno di un certo numero di risorse per svolgere il suo compito:
 - CPU, memoria, files e dispositivi di I/O
- Il sistema operativo è responsabile delle seguenti attività per la gestione dei processi:
 - creazione e terminazione dei processi
 - sospensione e ripristino dei processi
 - meccanismi per:
 - sincronizzazione di processi
 - comunicazione tra processi
 - gestione dello stallo

Gestione della memoria centrale

- La memoria centrale è costituita da una sequenza di celle ognuna con un proprio indirizzo
 - E' uno spazio per conservare dati facilmente accessibile per la CPU e i dispositivi di I/O.
- La componente del sistema operativo che si occupa della gestione della memoria principale è responsabile di:
 - Tenere traccia di quali parti della memoria sono correntemente usate e da chi
 - Allocare e deallocare lo spazio di memoria
 - Caricare processi inattivi quando la memoria si rende disponibile

Gestione dei File

- Un file è una collezione di informazioni in memoria secondaria
 - generalmente contiene dati o programmi (in formato sorgente o oggetto).
- Le directory sono file particolari
- La componente della gestione dei file del sistema operativo è responsabile delle seguenti attività:
 - Creazione e cancellazione di file
 - Creazione e cancellazione di directory
 - Supporto di primitive per manipolare file e directory
 - Mapping dei file sui dispositivi di memoria secondaria

Gestione dell'I/O

Il sistema di gestione dell'I/O consiste di:

- Un sistema di buffer-caching
- Una interfaccia generale per i driver dei dispositivi.
- Driver per dispositivi hardware specifici.

Gestione della memoria secondaria

- La maggior parte dei computer usano i dischi come dispositivi di memoria secondaria per contenere dati e programmi
 - Permettono di accedere ai file in modo diretto (RAM) e abbastanza rapido per una interazione efficace con utente/processi
- Per la gestione della memoria secondaria il sistema operativo si occupa di:
 - allocazione della memoria
 - gestione della memoria libera
 - scheduling del disco

Accesso alla rete

- In un sistema distribuito i calcolatori sono connessi attraverso una rete di comunicazione
 - L'accesso alla rete permette:
 - Condivisione di risorse
 - Prestazioni più alte
 - Maggiore disponibilità di dati
 - Maggiore affidabilità
- Le comunicazioni sono basati su *protocolli*
 - L'accesso alla rete avviene mediante moduli di programma che implementano le regole di tali protocolli

Sistema di Protezione

- Fornisce meccanismi per controllare l'accesso di processi e utenti alle risorse del sistema e/o degli utenti.
- I meccanismi di protezione devono:
 - distinguere tra usi autorizzati e usi non autorizzati
 - Specificare i controlli che devono essere imposti
 - fornire i mezzi per l'attuazione della protezione
- L'hardware permette di distinguere due modi di esecuzione
 - Utente
 - Sistema

Interprete comandi

- Il sistema operativo riceve continuamente comandi tramite istruzioni di controllo che riguardano:
 - creazione e gestione di processi,
 - gestione dell'I/O
 - gestione della memoria secondaria e accesso al file system
 - gestione della memoria centrale
 - protezione
 - Networking
- Il programma che legge e interpreta i comandi viene chiamato *interprete comandi* o *shell* (in UNIX)
 - ricevere il comando
 - ne controlla la correttezza
 - gestisce la sua esecuzione da parte dei diversi livelli del sistema
 - mostra i risultati

System Call e Programmi di Sistema

- Le **System call** forniscono l'interfaccia di programmazione tra programmi e SO
 - Application Programming Interface (API)
 - Generalmente a basso livello, in alcuni casi con sintassi simile ai linguaggi di alto livello (es. C e C++)
 - Le system call offerte da un sistema operativo riguardano le diverse funzionalità del sistema, come:
 - Controllo dei processi
 - Gestione dei file
 - Gestione dei dispositivi di I/O
 - Informazioni sul sistema
 - Comunicazioni
- I **Programmi di Sistema** forniscono un ambiente conveniente per lo sviluppo e l'esecuzione di applicazioni
 - permettendo l'accesso alle funzioni delle system call
 - forniscono supporto ai linguaggi di programmazione
 - si occupano del caricamento e dell'esecuzione dei programmi
- La maggior parte degli utenti conosce il sistema operativo tramite i programmi di sistema e non tramite le system call

Organizzazione di un Sistema Operativo

- Come sono organizzate le varie componenti di un S.O.?
- Quali sono le modalità di interazione tra esse?
 - Sistemi monolitici
 - Sistemi modulari
 - Sistemi a livelli
 - Sistemi a microkernel
 - Modello cliente-servitore

Sistema Monolitico

Il sistema operativo è costituito da un unico programma contenente un insieme di procedure, che realizzano le varie componenti

l'interazione tra le componenti avviene mediante il meccanismo di chiamata a procedura

Sistemi modulari

Suddivisione del sistema in moduli, ciascuno destinato a fornire una determinata funzione

INTERFACCIA

Specificità delle funzionalità offerte dal modulo

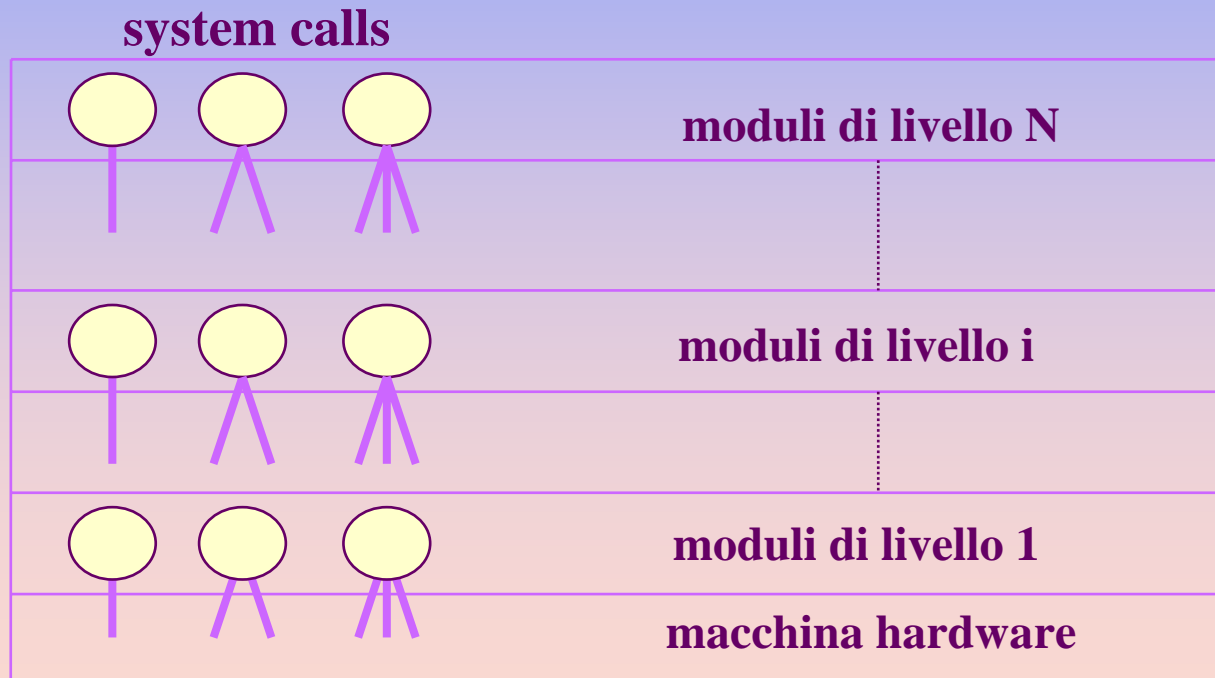
CORPO

Realizzazione delle funzionalità non visibile all'esterno

Sistemi a livelli

Le funzioni del S.O. sono organizzate a livelli gerarchici

In ogni livello sono usate le funzionalità dei livelli sottostanti

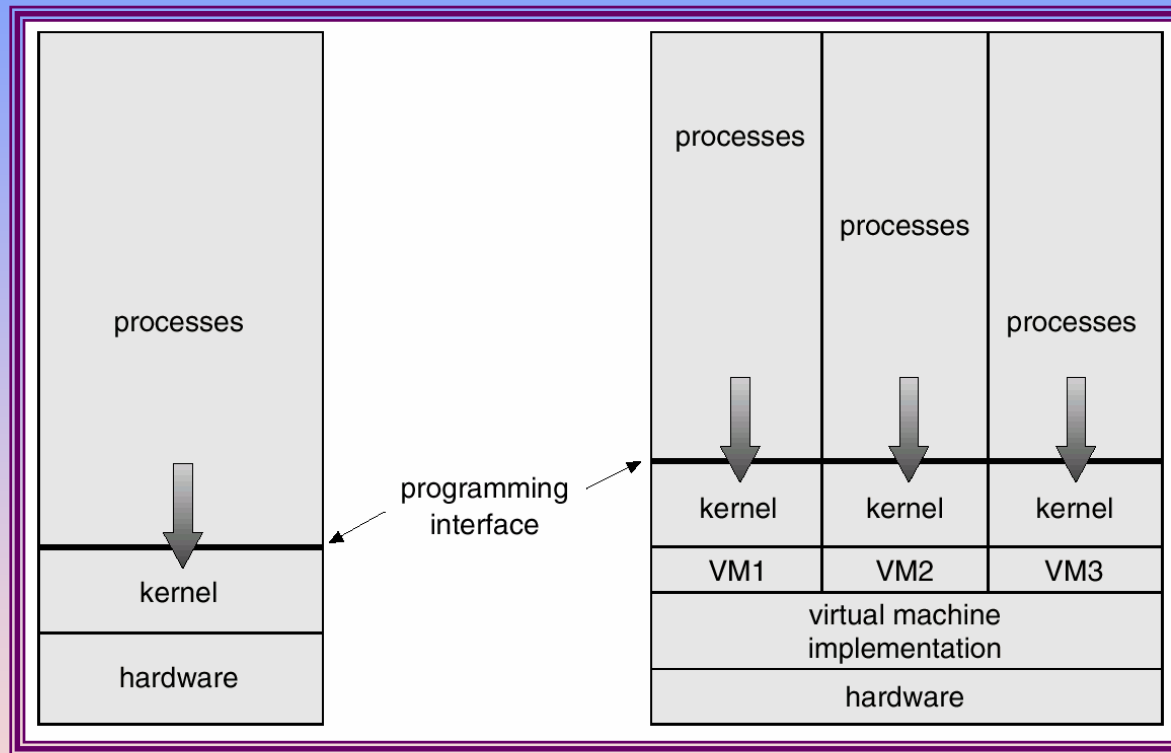


Macchine Virtuali

- Replicazione (virtualizzazione) dell'hardware
 - L'hardware e il nucleo sono considerati come fossero un unico elemento.
 - Una macchina virtuale fornisce una identica interfaccia a diverse architetture hardware.

- Il sistema operativo crea l'impressione di diversi processi ognuno in esecuzione su una propria macchina
 - Le risorse fisiche di un computer sono condivise per creare le macchine virtuali.
 - Lo scheduling della CPU crea l'impressione che ogni utente abbia il proprio processore.
 - La memoria centrale è suddivisa fra i vari processi
 - Si possono avere virtualizzazioni di dispositivi di I/O

Organizzazione a macchine virtuali

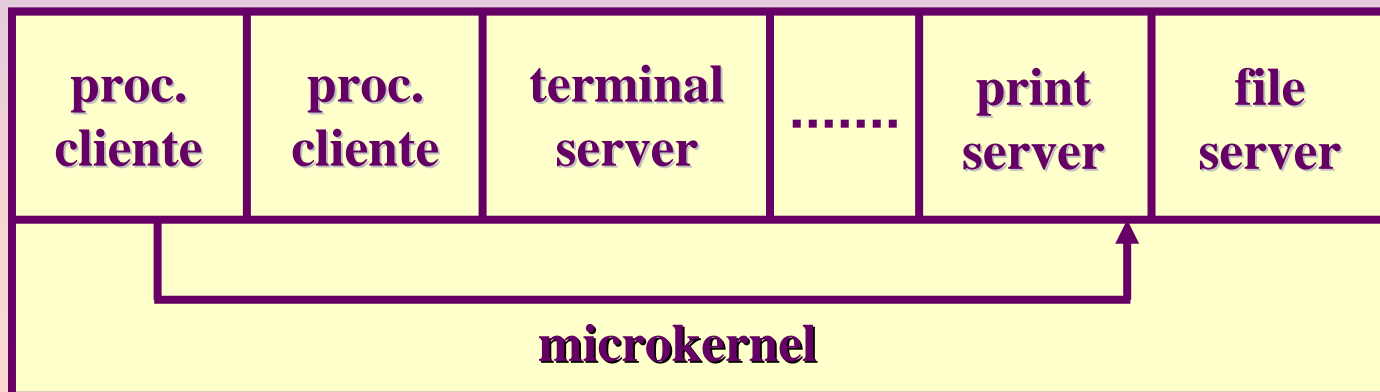


Senza Macchina Virtuale

Macchina Virtuale

Struttura a microkernel

- Due componenti del S.O. per ogni risorsa:
 - meccanismi per la gestione della risorsa
 - politiche di gestione
- Microkernel:
 - insieme dei meccanismi (stato privilegiato)
- Tutte le politiche sono realizzate come normali processi (stato utente).

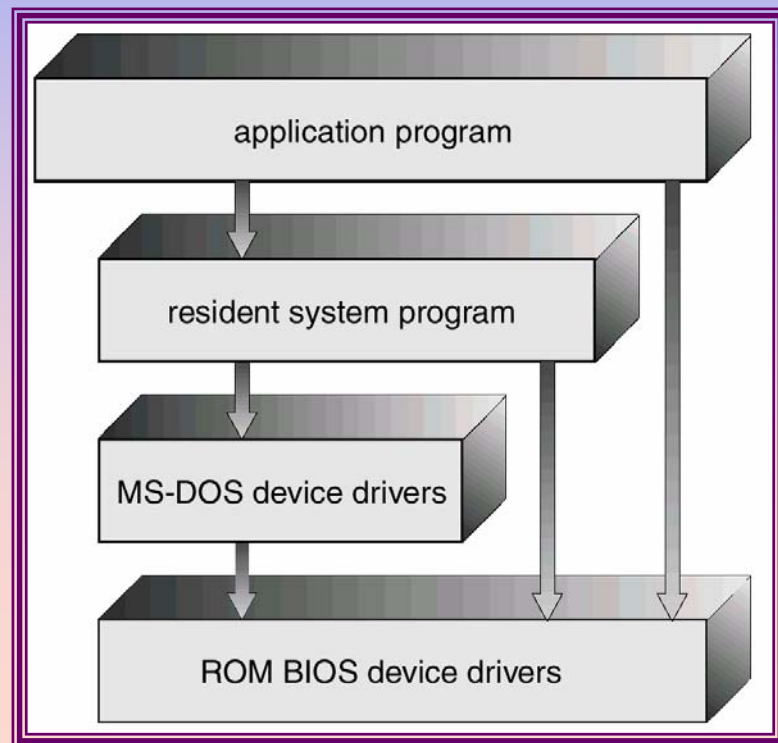


Modello cliente-servitore

- Un processo utente (**client**) richiede un servizio (es. lettura di un file) ad un processo di S.O. (**server**)
- Al termine del lavoro, il processo server rispedisce al client un messaggio di risposta
- Client e server operano nello spazio di utente

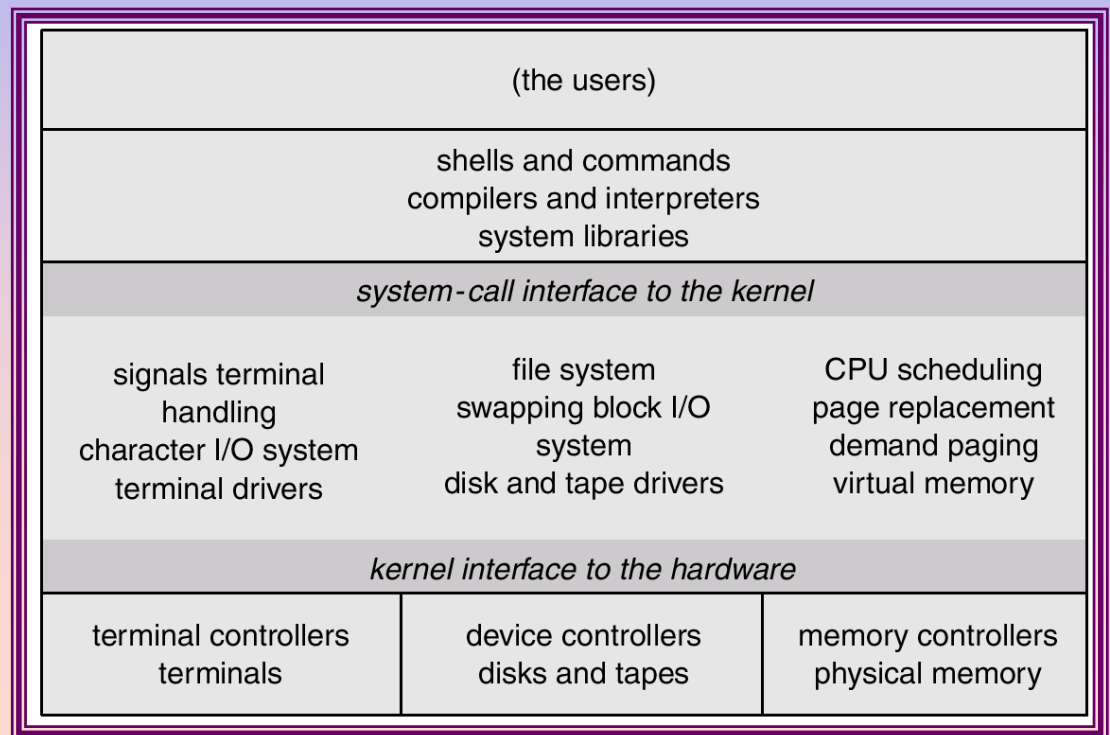
Struttura del sistema MS-DOS

- MS-DOS è stato ideato per fornire le funzionalità principali occupando poco spazio di memoria
 - originato dal Q-DOS (Quick and Dirty Operating System)
- Non è diviso in moduli.
- Le interfacce e i livelli funzionali non sono ben separati

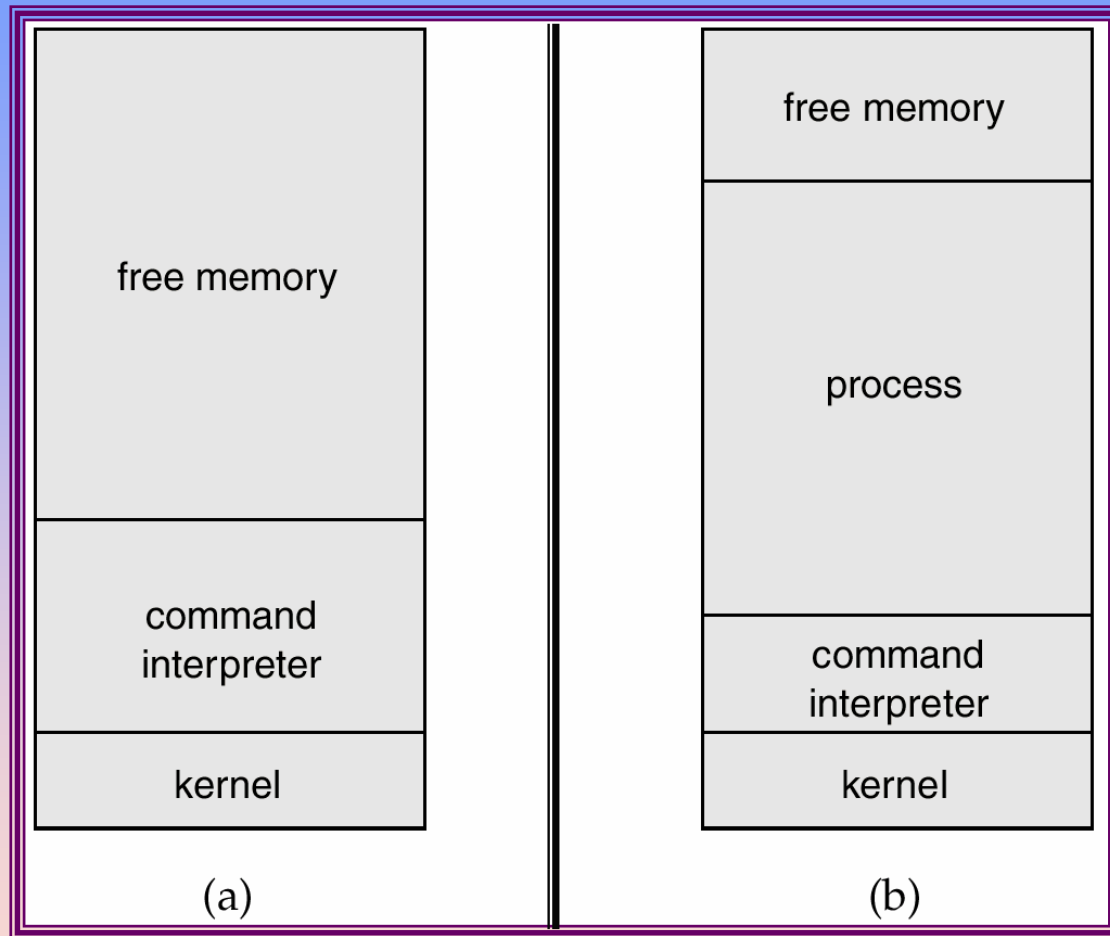


Struttura del sistema UNIX

- UNIX (Linux) consiste di due componenti principali.
 - I programmi di sistema
 - Il nucleo (kernel)
 - Tra il livello delle system call e l'hardware,
 - Fornisce il file system, lo scheduling della CPU, gestione della memoria, ed altre funzioni di base.



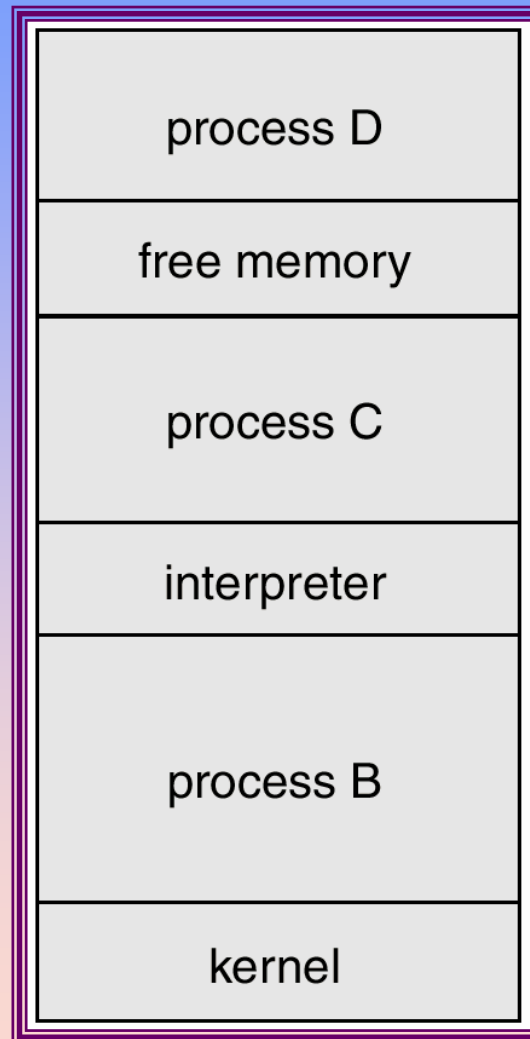
Esecuzione MS-DOS: un solo programma



System Start-up

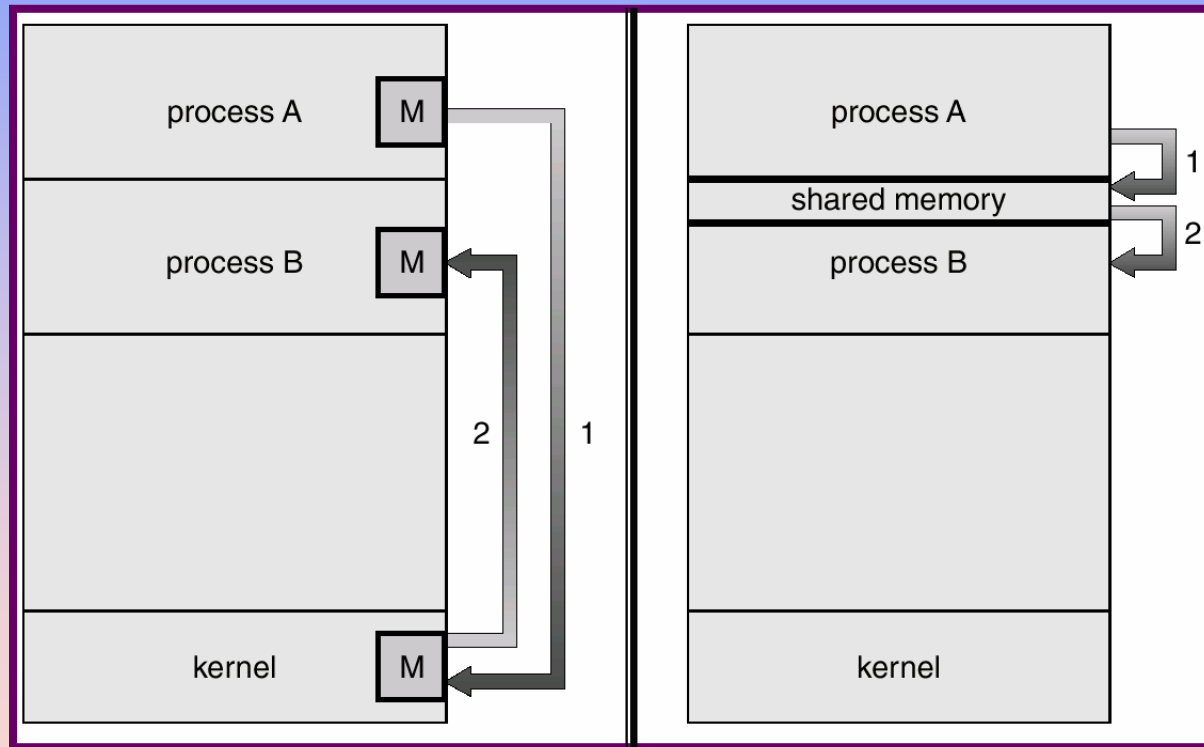
Esecuzione di un programma

UNIX: Esecuzione di più programmi



Modelli di comunicazione

- Le comunicazioni tra i processi possono avvenire usando lo scambio di messaggi o la memoria condivisa.



Scambio messaggi

Memoria condivisa

Java Virtual Machine (JVM)

- Macchina virtuale per l'esecuzione di programmi Java
 - Il programmi Java dopo la compilazione sono *architecture-neutral* e sono eseguibili dalla JVM
- La JVM consiste di
 - un class loader
 - un class verifier
 - un runtime interpreter
- I compilatori *Just-In-Time* (JIT) aumentano le prestazioni dei programmi Java.

Java Virtual Machine

