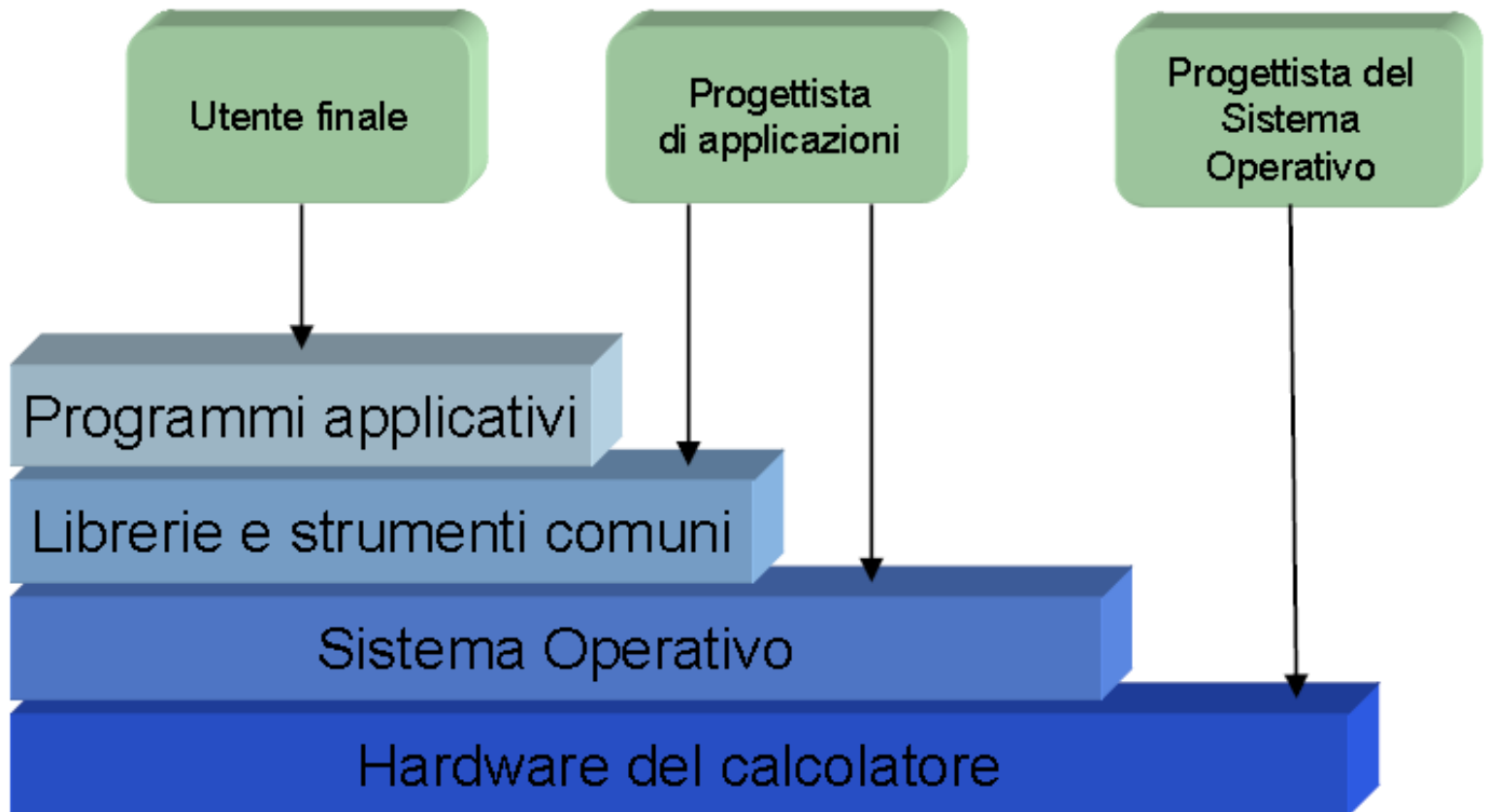


Sistema operativo

Definizione

- Parte del software che gestisce
 - I programmi applicativi
 - L'interfaccia tra il calcolatore e i programmi applicativi
 - Le funzionalità di base

Architettura a strati di un calcolatore



Generalità.

- All'accensione dell'elaboratore viene eseguito il programma di bootstrap
- il bootstrap è contenuto nel firmware dell'elaboratore, inizializza i dispositivi e carica il kernel del s.o.
- Il s. o. può funzionare sia in modalità utente che in modalità sistema.
 - Nel primo caso, è l'utente a definire il codice di comando
 - Nel secondo caso, è il s.o ad agire automaticamente secondo direttive prestabilite
 - In genere, il s.o. lavora secondo entrambe le due modalità

Classificazione dei s.o

- Sistemi tradizionali:
 - Monoprocessore, utilizzano un solo processore
 - Multiprocessore, utilizzano più unità di esecuzione
 - Cluster, più elaboratori completi collegati tra loro per svolgere attività comuni
 - Sistemi distribuiti, formati da più elaboratori eterogenei posti a distanza e interconnessi attraverso mezzi trasmissivi. Un sistema operativo in rete utilizza due possibili architetture: client-server o peer-to-peer. Nel primo, il client interroga il server per poter accedere al data base. Nel secondo caso, gli utenti decidono quali risorse condividere, sono tutti alla pari

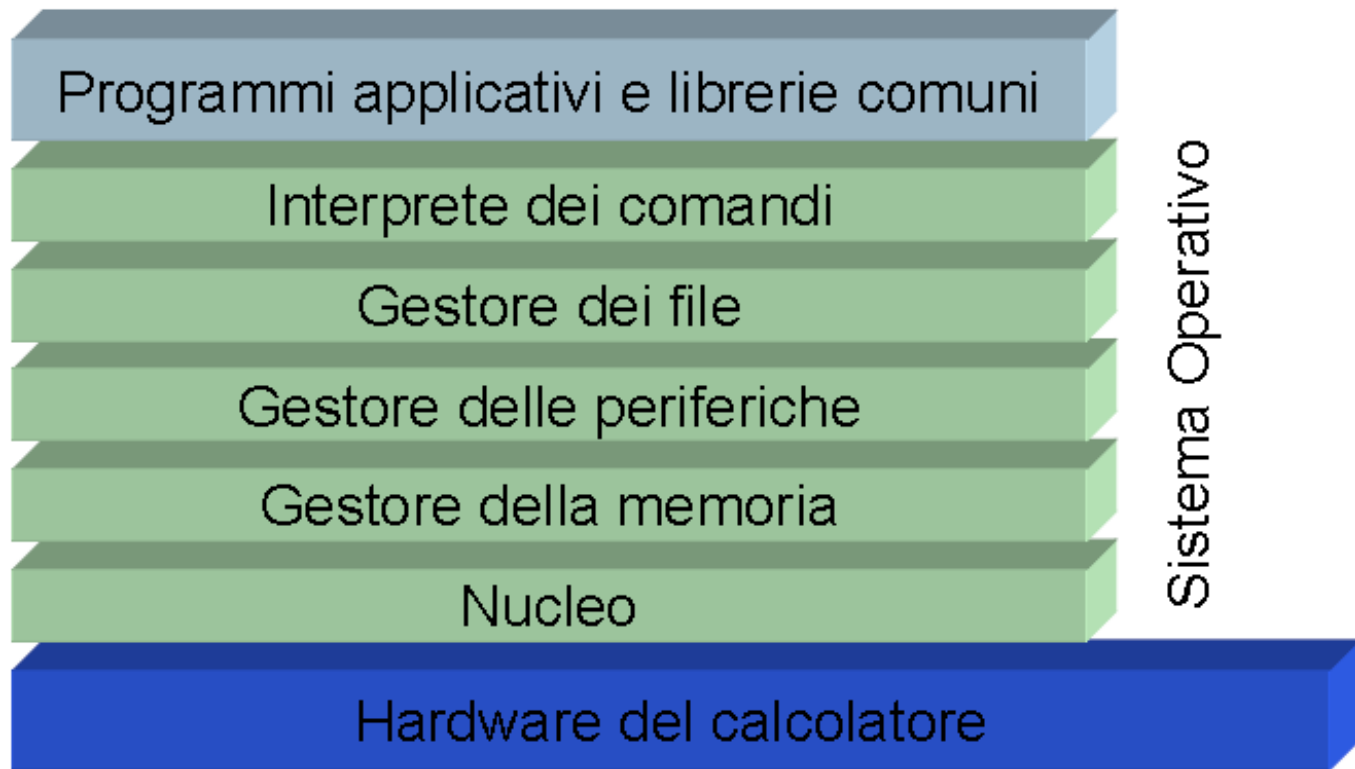
Classificazione dei s.o

- Sistemi real time, sono quelli utilizzati per il controllo di un processo; il tempo di intervento deve essere un tempo utile, adeguato al processo.
- Sistemi per la multimedialità, gestiscono file audio e video
- Sistemi per palmari, sistemi dalle dimensioni ridotte; comprendono i PDA (Personal Digital Assistant), i palmari, i-phone, telefoni digitali. La tecnologia veloce dei 3G unisce prodotti diversi tra loro: telefono, navigatore satellitare, fotocamera, internet. I sistemi operativi devono gestire una piccola memoria virtuale con risparmio energetico e piccoli schermi.

Architettura a strati del sistema operativo

- I sistemi operativi hanno una struttura complessa
- possono essere schematizzati come una gerarchia di macchine virtuali

Livelli del sistema operativo



Servizi di un s.o.

- Interfaccia con l'utente
- Esecuzione di un programma: carica in memoria le applicazioni e le esegue
- Gestione dei processi
- Assegnazione delle risorse: quando vengono eseguiti i processi, il s.o. ha il compito di assegnare le risorse di input o output
- Gestione memoria
- Gestione file system
- Sicurezza e protezione
- Gestione I/O

Interfaccia con l'utente

- Può essere in due modalità:
 - Interprete comandi o shell: preleva i comandi digitali dall'utente e li esegue. Il più delle volte, l'interprete viene considerato parte integrante del kernel. È possibile implementare i comandi in due modi:
 - Il codice del comando può essere contenuto all'interno dell'interprete stesso; l'esecuzione equivale ad un salto ad una sezione del codice che invoca le chiamate opportune con parametri opportunamente impostati
 - I comandi sono implementati separatamente all'interno del file di sistema; l'interprete si limita a leggere il comando, carica in memoria il nome del file corrispondente. In questo modo si alleggerisce lo shell. Si possono aggiungere o togliere i comandi.

La shell non deve essere unica. Il sistema Unix eLinux possono scegliere diversi interpreti.

Interfaccia con l'utente

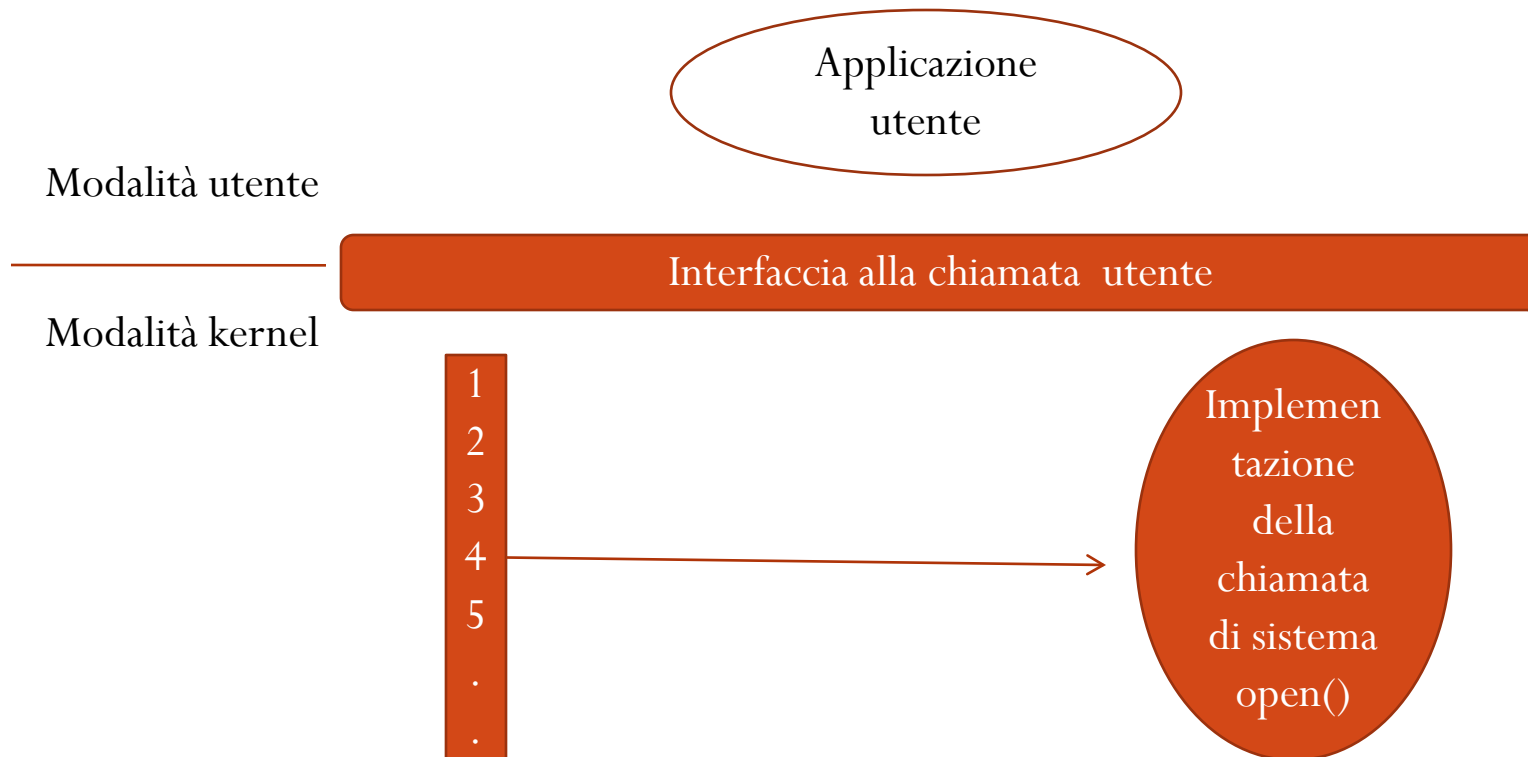
- Interfaccia grafica: la maggior parte dei s.o è dotato di una GUI, Graphics User Interface. Il primo è stato il sistema Mac Os
- L'origine risale agli anni 70 presso i laboratori della xerox di Palo Alto, USA
- Notevole impulso è stato dato dagli ambienti grafici open-source compatibili sia con Linux che con Unix

Chiamate di sistema

- Sono l'interfaccia tra i processi e il sistema operativo
- Ogni tasto premuto sulla tastiera è una chiamata a s.o
- Il programmatore utilizza una interfaccia per la programmazione detta API, Application Programming Interface
- L'API semplificano molto i compiti del programmatore perché mette a disposizione un insieme di funzioni che invocano chiamate al s.o. semplificando la progettazione; il programma può essere inoltre utilizzato su qualsiasi sistema che utilizza la stessa API (es API Win32, API Posix...)
- Ogni funzione all'interno di una libreria sono una chiamata a sistema e identificate con un numero

Chiamate di sistema

- Riguardano principalmente: controllo di processo, gestione dei file, gestione dei dispositivi, gestione delle informazioni, comunicazione tra i processi



Concetto di processo e di risorse

- Processo: programma in fase di esecuzione cui si associa il valore del program counter, dei registri del processore, della memoria e dello stack
- Il programma è una sequenza di azioni che il processore deve svolgere
- Il processo è un'entità dinamica, il programma è un'entità dinamica
- Si può lanciare ripetutamente lo stesso programma, viene comunque utilizzato lo stesso codice
- Durante la sua esecuzione, ogni processo può generare altri processi detti processi figli e il processo che li ha generati, padre
- Ogni processo è caratterizzato da un numero detto PID, *Process Identifier*
- Se i processi attivi contemporaneamente sono tanti, la CPU sarà utilizzata in una gran percentuale
- Ogni processo viene caricato nella memoria RAM
- Le risorse dei processi sono sia la parte hardware che software

Il nucleo

- Il nucleo o kernel è il responsabile della gestione dei processori
- Il nucleo gestisce un solo processore
- Se il nucleo gestisce un solo programma alla volta, si parla di uniprogrammazione
- I sistemi operativi attuali gestiscono più processi alla volta, la multiprogrammazione
 - Ogni processo non risente della presenza degli altri
- Il nucleo può gestire anche più processori

I thread

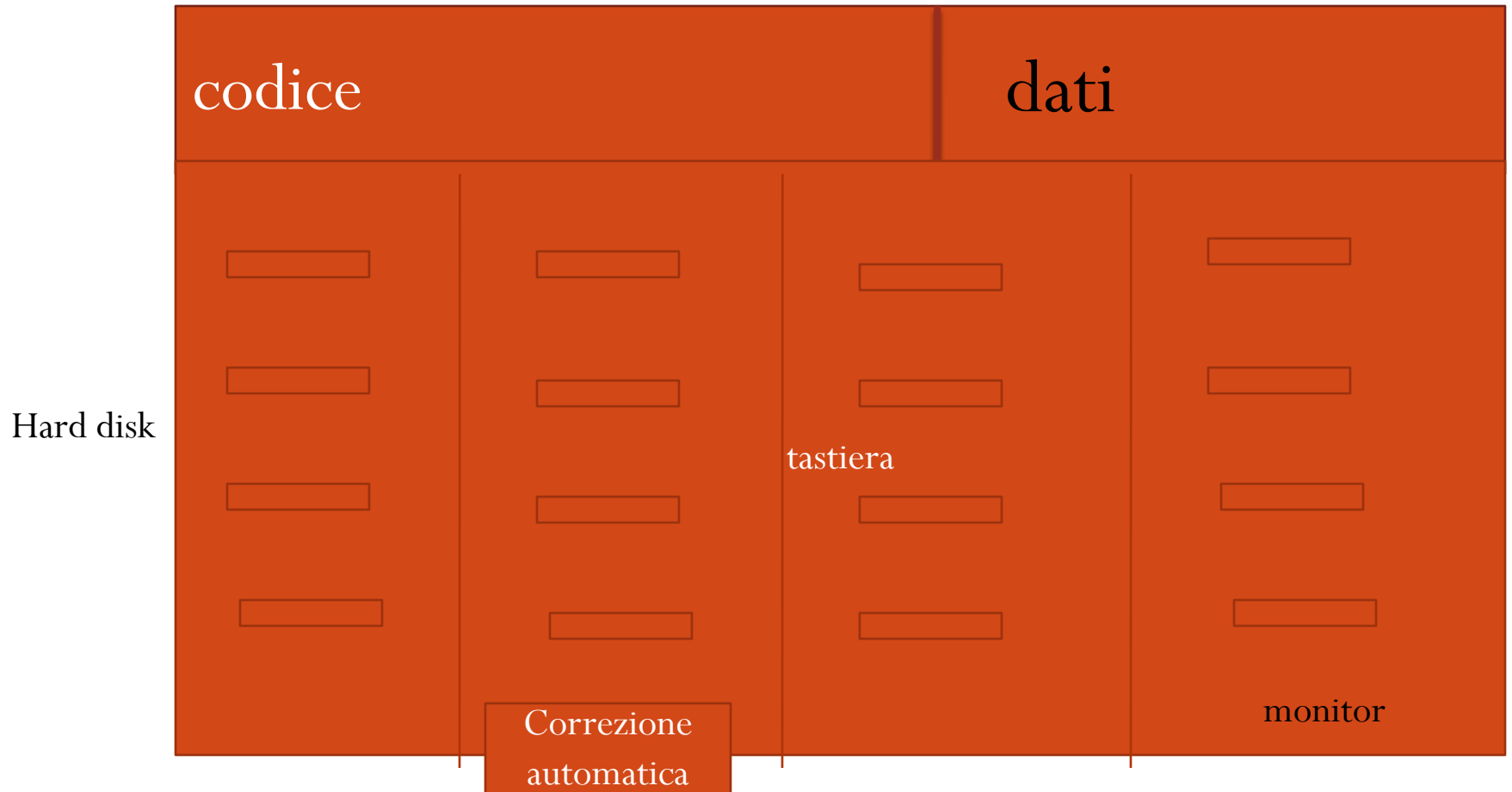
- Piccola unità base di esecuzione del processore; comprende un program counter, uno stack, un identificatore, un set di registri
- Un processo tradizionale segue un solo percorso; i processi attuali seguono più percorsi
- I thread di uno stesso processo condividono lo stesso codice e quindi, lo stesso spazio di indirizzamento
- Es. un browser utilizza più thread: uno per scaricare immagini, uno per i suoni... Un elaboratore testi utilizza un thread per visualizzare testi, uno per salvarli periodicamente, un altro per acquisire i dati da tastiera.
- Conviene utilizzare più thread di un solo processo che più processi insieme per velocizzare il processore

Thread

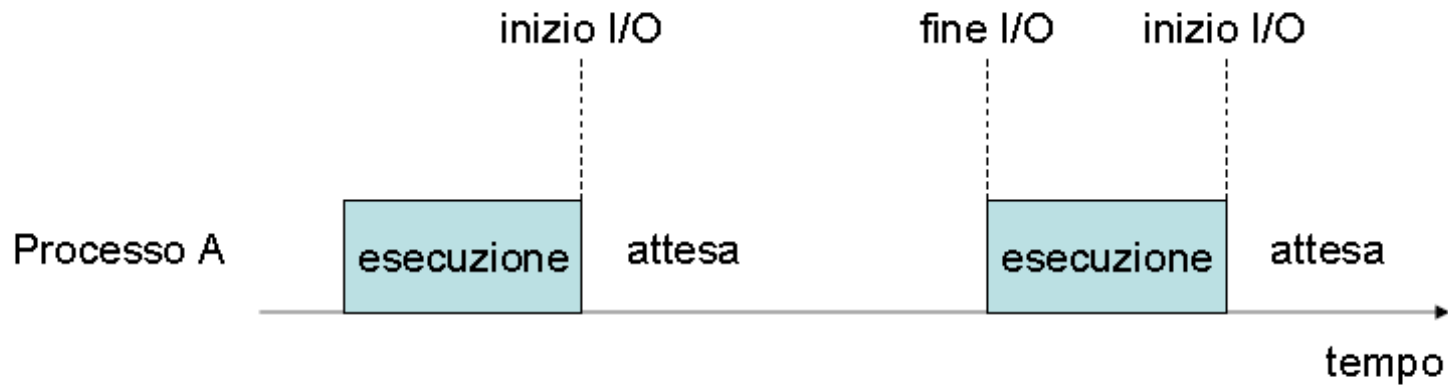
- Un processo ad un solo thread



1 thread per un processo di videoscrittura



Uniprogrammazione



Multiprogrammazione

