

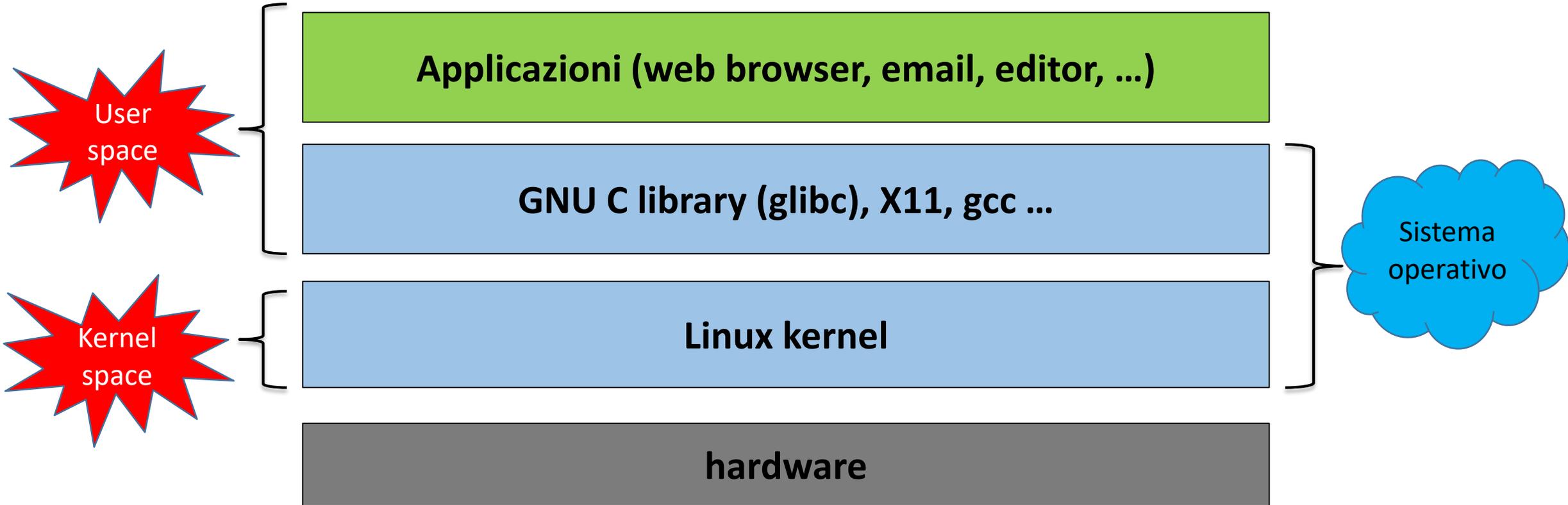
Corso «Sistemi Operativi»

AA 2020-2021

Marco Tessarotto

Compiti del kernel

OS Linux



Compiti del kernel

- Il kernel è il programma che gestisce ed assegna le risorse del calcolatore (CPU, RAM, spazio su disco, periferiche...)
- Schedulazione dei processi
- Creazione e terminazione dei processi
- Gestione della memoria fisica (RAM)

- Fornitura di un file system («sistema di file», «sistema di archivi»)
- Accesso alle periferiche
- Networking
- System Call Interface (o API)

Schedulazione dei processi – Process scheduling

- Un computer dispone di una o più unità di elaborazione centrale (CPU), che eseguono le istruzioni dei programmi.
- Processo: istanza di un programma in esecuzione
- Come altri sistemi UNIX, Linux è un **sistema operativo multitasking preventivo (preemptive)**
- **Multitasking** : più processi (cioè programmi in esecuzione) possono risiedere simultaneamente in memoria e ognuno può ricevere (a turno) l'uso della/e CPU.
- **Preemptive**: le regole che governano quali processi ricevono l'uso della CPU e per quanto tempo sono determinate dallo schedulatore di processi dei processi del kernel (piuttosto che dai processi stessi)

Creazione e terminazione dei processi - process management

- Un processo esistente invoca una system call per chiedere al kernel di eseguire un programma (viene specificato il file contenente il codice macchina)
- il kernel carica il programma in memoria e alloca la memoria
- Il kernel prepara codice, dati, stack del nuovo processo
- Il kernel ha completato la creazione del nuovo processo; poi è compito dello schedulatore dei processi fare eseguire il nuovo processo

Creazione e terminazione dei processi - process management

- Il kernel fornisce al nuovo processo le risorse (ad esempio, CPU, memoria e accesso ai file) di cui ha bisogno per funzionare
- Una volta che un processo ha completato l'esecuzione, il kernel garantisce che le risorse precedentemente assegnate siano liberate per un successivo riutilizzo da parte dei processi successivi

Gestione della memoria fisica (RAM)

Memory management

- La memoria RAM è una risorsa limitata, da suddividere tra i programmi in esecuzione in modo equo ed efficiente
- Linux adotta la gestione della memoria virtuale, vantaggi:
 - I processi sono isolati uno dall'altro e dal kernel
 - Solo parte di un processo richiede di stare in memoria RAM
- La memoria può essere allocata ad un processo / deallocata da un processo (ad es. quando termina)
- Il gestore della memoria tiene traccia di quale memoria è utilizzata e da chi

Fornitura di un file system

- un file system controlla il modo in cui i dati vengono archiviati e recuperati nella memoria di massa (non volatile)
- Creare/cancellare/leggere/scrivere file
- Gestione dello spazio libero

Accesso alle periferiche – access to devices

- Il kernel tramite i suoi driver di periferica fornisce ai processi una interfaccia che standardizza e semplifica l'accesso alle periferiche
- il kernel arbitra (regola) l'accesso da parte dei processi alle periferiche

Networking

- Il kernel trasmette e riceve messaggi di rete (pacchetti) per conto dei processi utente.
- Questa attività include l'instradamento dei pacchetti di rete al sistema di destinazione (cioè svolge un servizio per gli altri calcolatori/dispositivi connessi in rete)

Interfaccia delle chiamate di sistema - System Call Interface

- I processi richiedono al kernel di svolgere azioni attraverso le system call (punti di ingresso nel kernel)
- L'insieme delle system call definisce la system call interface

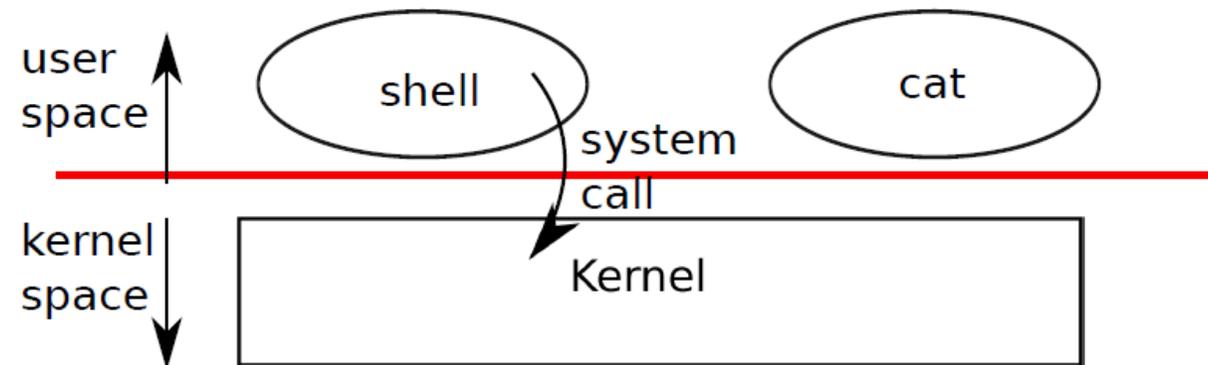


Figure 0-1. A kernel and two user processes.

Linux OS – sistema operativo multiutente

- i sistemi operativi multiutente: forniscono agli utenti l'astrazione di un computer privato virtuale;
- ogni utente può accedere al sistema e operare indipendentemente dagli altri utenti.
- ogni utente ha il proprio spazio di archiviazione su disco (home directory).
- ogni utente può eseguire i suoi programmi, ognuno dei quali ottiene una quota della CPU e opera nel proprio spazio di memoria virtuale; questi programmi possono accedere in modo indipendente ai dispositivi e trasferire informazioni sul network.
- Il kernel risolve potenziali conflitti nell'accesso alle risorse hardware, quindi utenti e processi sono generalmente inconsapevoli dei conflitti.