

ONE PIECE



3 - ENIES LOBBY

Enies Lobby

Nico Robin

Ohara

Imperatori

Gen. - Feb. - Mar.

Introduzione

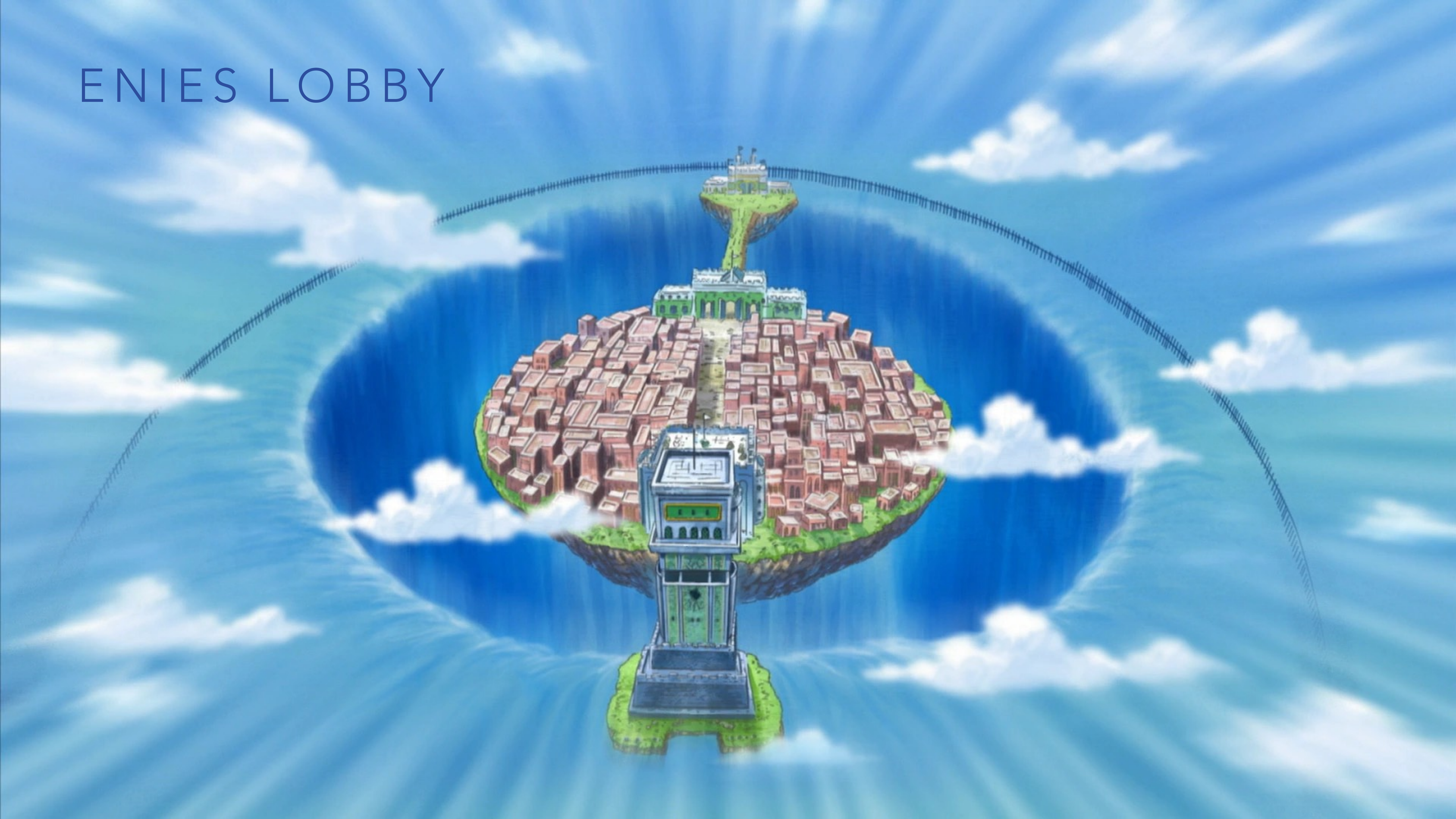
Storia

Architetture

Processi



ENIES LOBBY



Introduzione ai Sistemi Operativi

Sistemi Operativi

Introduzione ai Sistemi Operativi

Argomenti del capitolo

- Introduzione
- Modalità
- Struttura
- Cos'è e cosa fa

Introduzione

Introduzione ai Sistemi Operativi

Introduzione ai Sistemi Operativi

Introduzione

Dispositivi moderni

- Esecuzione di più App
- Esecuzione di programmi che occupano più memoria di quella disponibile
- Gestione di file in locale e remoti
- Gestione multi utente

Modalità

Introduzione ai Sistemi Operativi

Introduzione ai Sistemi Operativi

Modalità

User Mode

Numero limitato di istruzioni macchina* disponibili

Kernel Mode

Tutte le istruzioni macchina* disponibili

- Il sistema operativo e le applicazioni girano entrambe in user mode, mentre solo i driver girano in kernel mode.
- PSW (Program Status Word): registro della CPU in cui c'è un bit che serve per identificare la modalità.



BUG

Il sistema operativo e le applicazioni girano entrambe in user mode, mentre solo i driver girano in kernel mode.

Correzione:

Il sistema operativo gira in kernel mode, il resto in user mode



Introduzione ai Sistemi Operativi

Modalità

GUI (Graphical User Interface)

- La GUI è una parte del kernel
- Non è indispensabile.
- Esegue in modalità user.
- Per interagire con il SO (avviare/fermare programmi)
- Aiuta a svolgere dei compiti del SO (es. password)

OS MS Windows 1.0 GUI (1985)

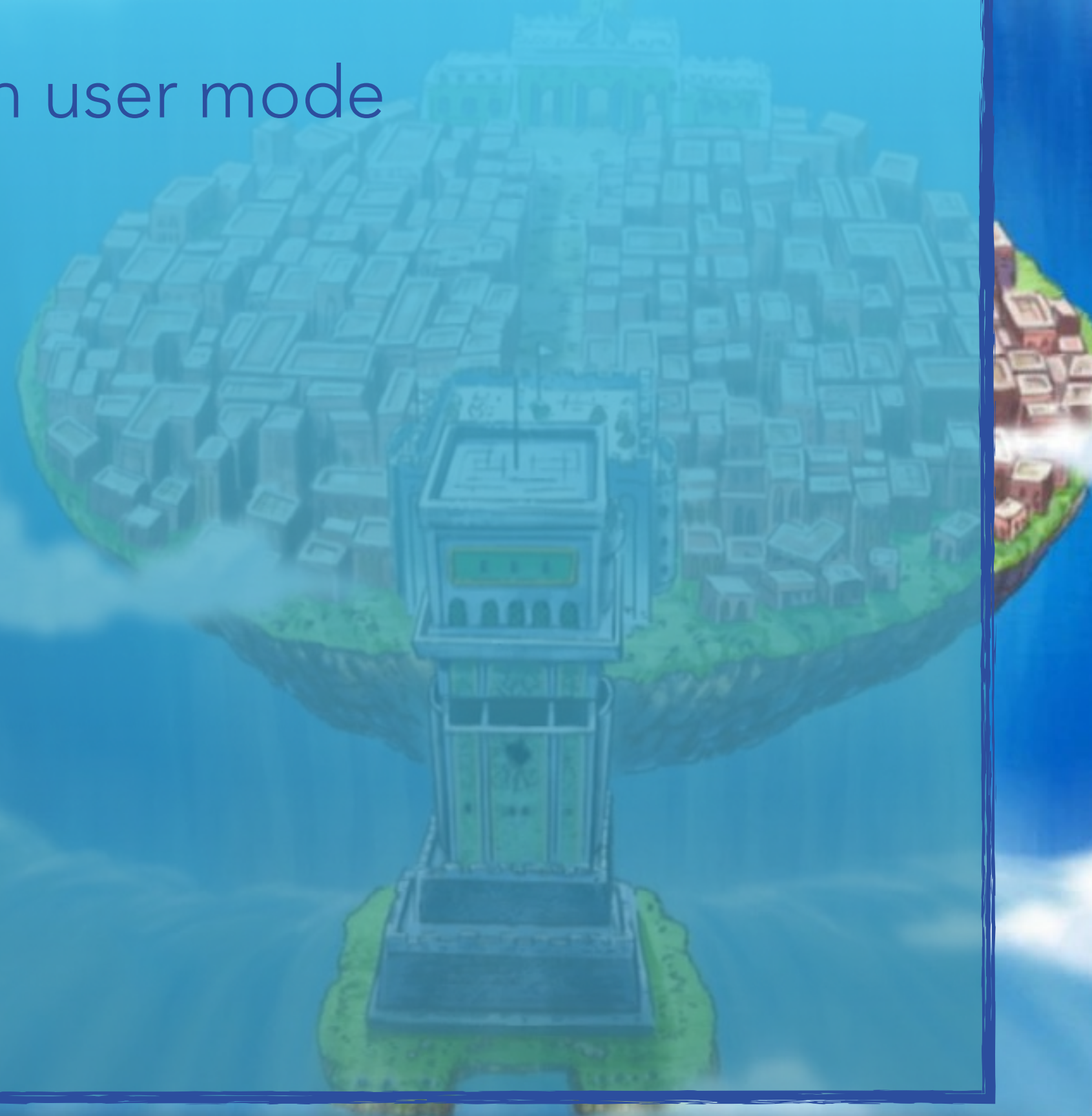


BUG

La GUI è una parte del kernel

Correzione:

Non è indispensabile e gira in user mode



ANSIA

Quanto vale in decimale il numero esadecimale 15?



ANSIA

Quanto vale in decimale il numero esadecimale 15?

Risposta:

21



Struttura

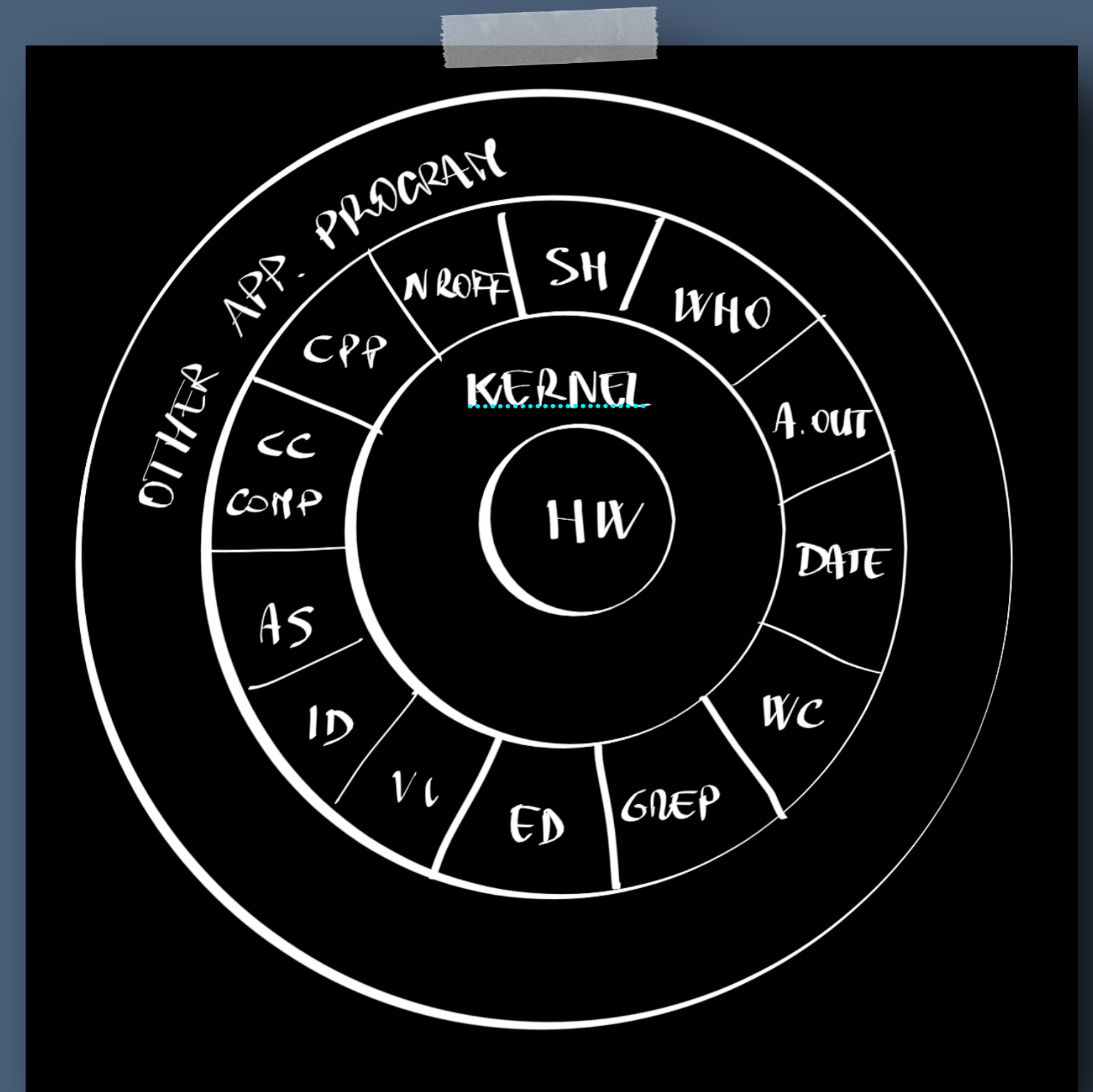
Introduzione ai Sistemi Operativi

Introduzione ai Sistemi Operativi

Struttura

Unix

Le funzioni che svolge Unix vengono svolte da tutti i SO.



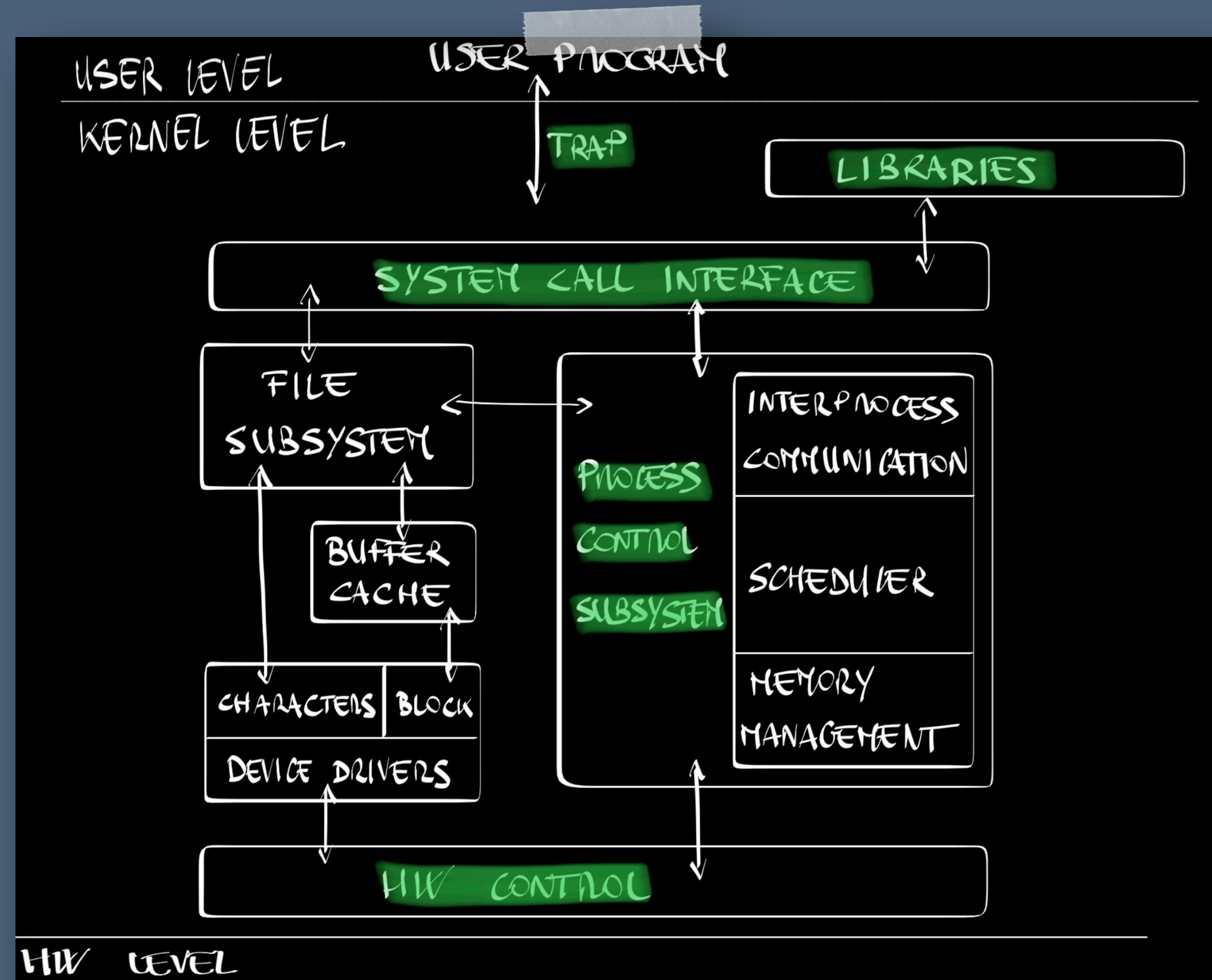
Unix V (1986) - No GUI, solo shell - Struttura monolitica

Introduzione ai Sistemi Operativi

Struttura

Le funzioni che svolge Unix vengono svolte da tutti i SO.

In verde ciò che vedremo



Introduzione ai Sistemi Operativi

Struttura

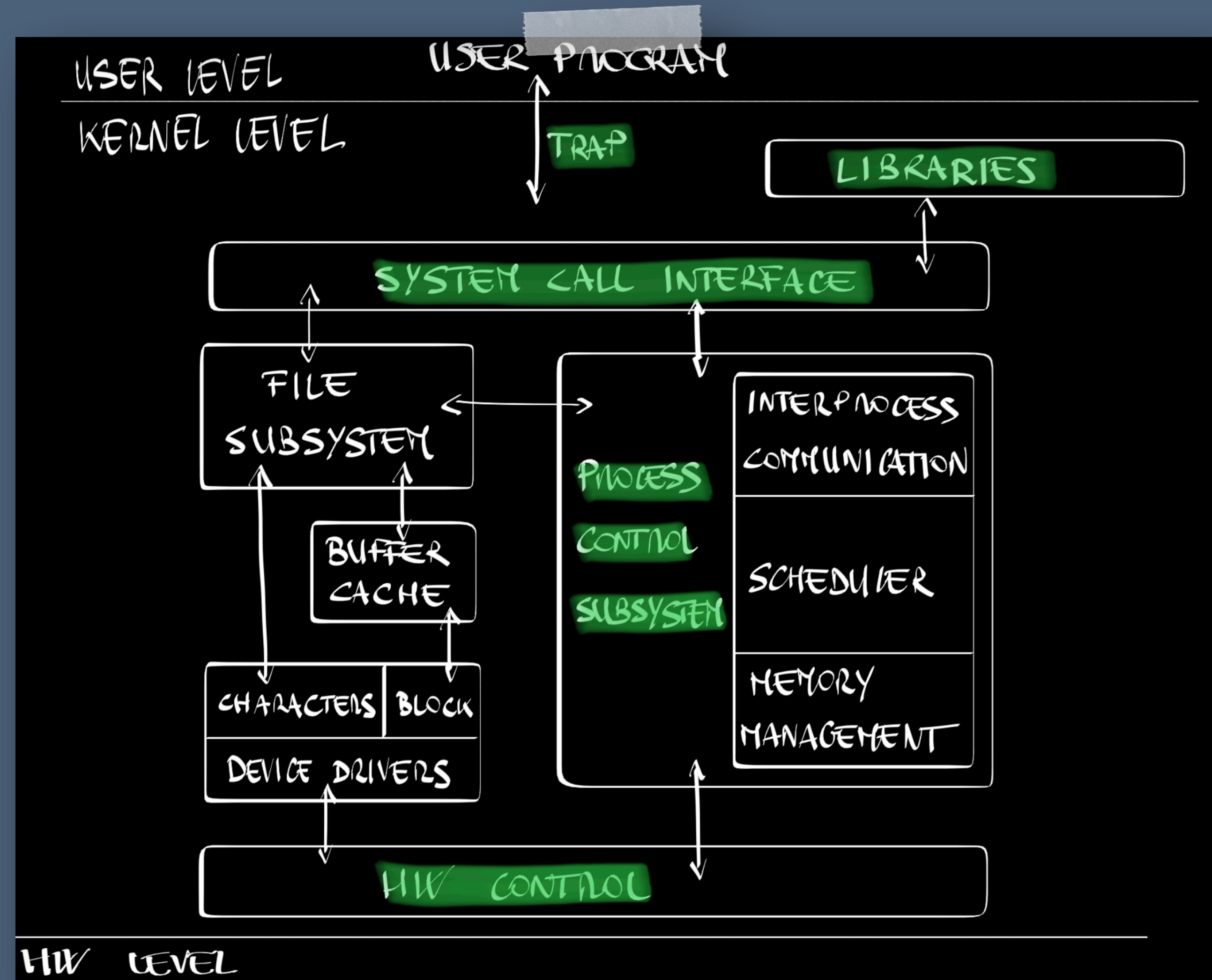
Hardware control

Comunicazione con l' hw mediante la lettura/scrittura dei registri del controllore e la gestione degli interrupt.

Device drivers

Controllo dei dispositivi, a blocchi e a caratteri.

(Non trattato)



Introduzione ai Sistemi Operativi

Struttura

Buffer cache

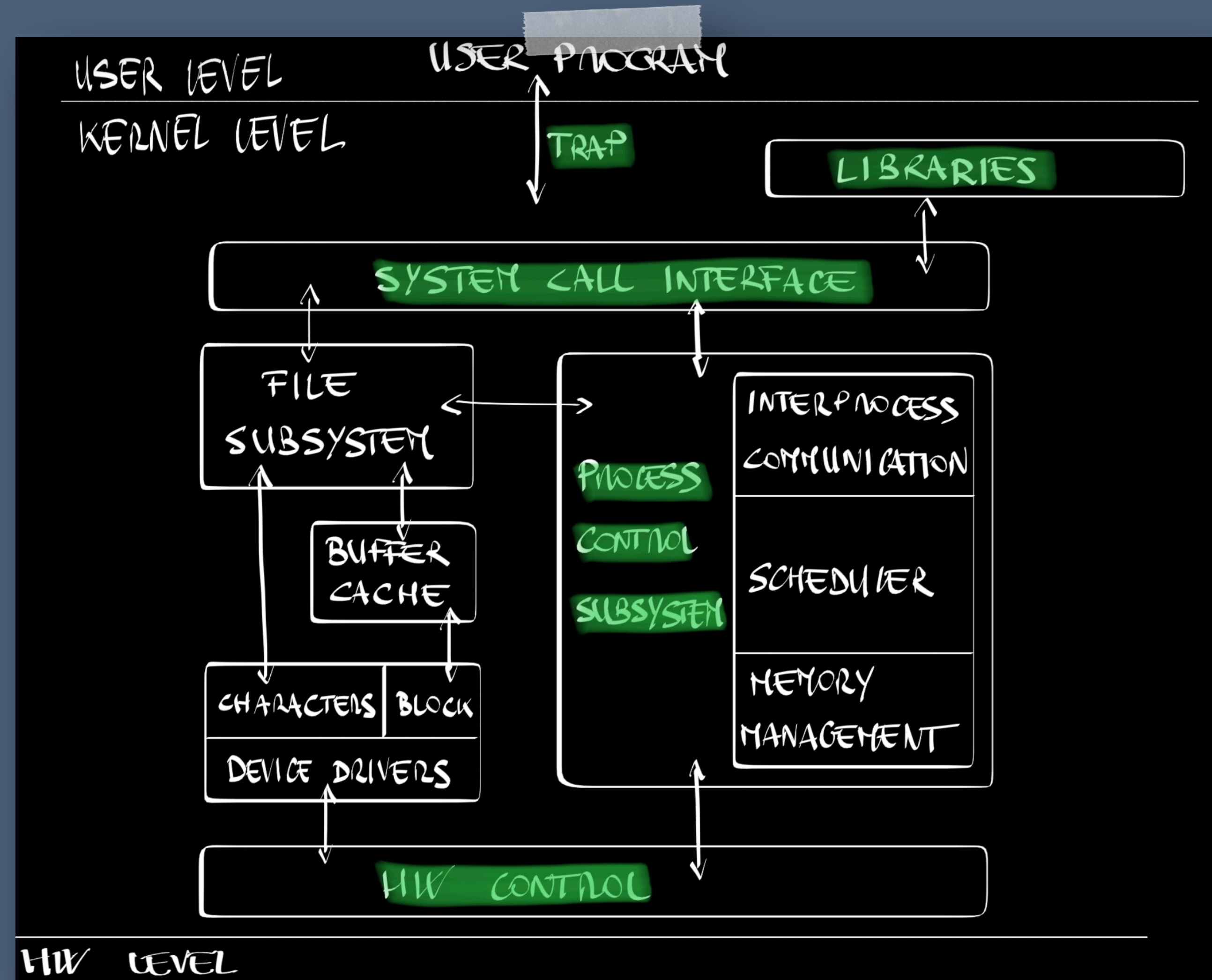
Cache in memoria per i dispositivi a blocchi, quali, per esempio, il disco.

(Non trattato)

File subsystem

Realizzazione e gestione file. In Unix l'interfaccia verso i dispositivi è uguale all'interfaccia verso i file.

(Non trattato)



Introduzione ai Sistemi Operativi

Struttura

Process control subsystem

Ambiente a processi:

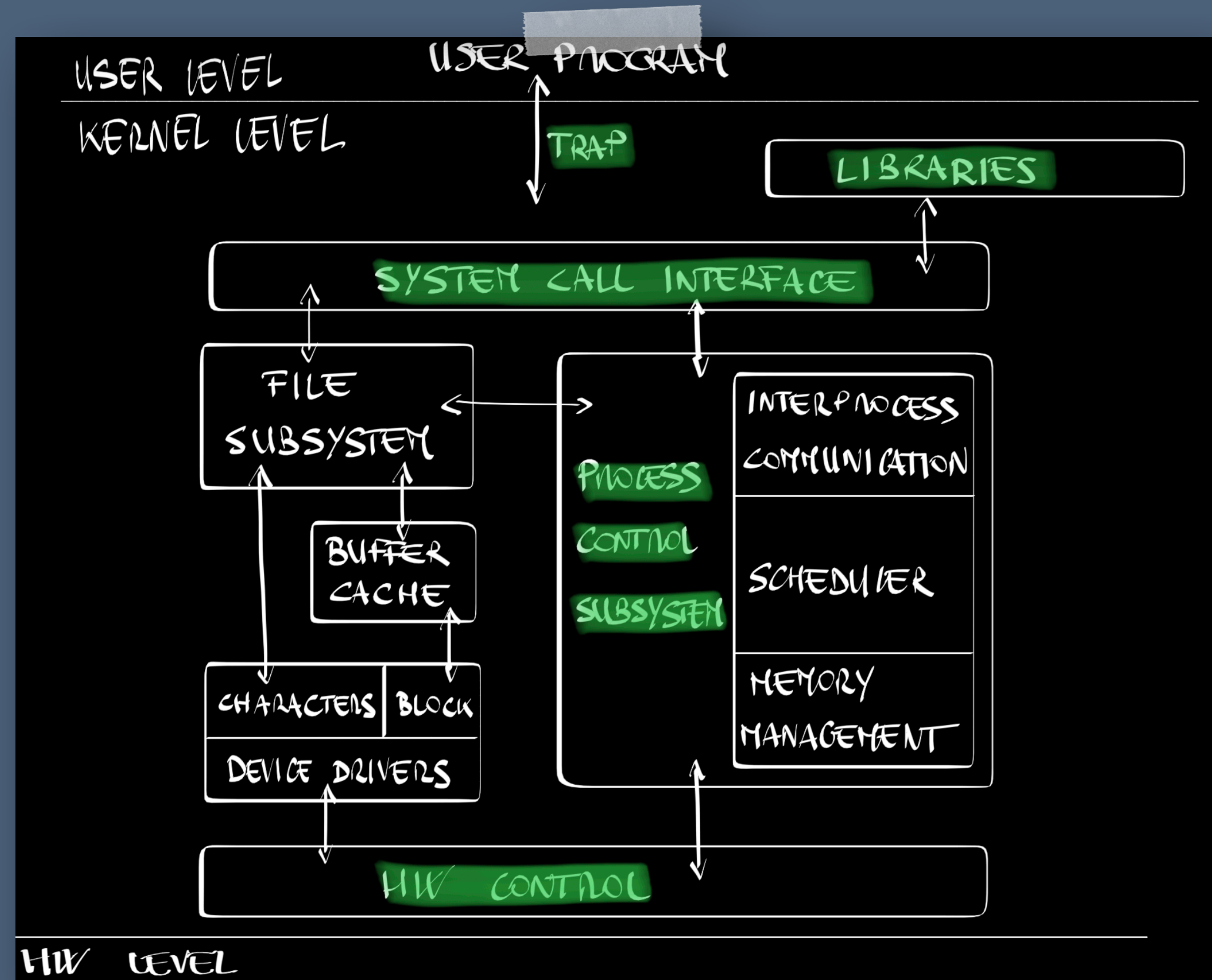
- Inter Process Communication: gestione comunicazioni tra processi.

(Capitolo Sincronizzazione)

- Scheduler: allocazione della CPU ai processi.

(Non trattato)

- Memory management: allocazione della memoria al SO ed ai processi.



Introduzione ai Sistemi Operativi

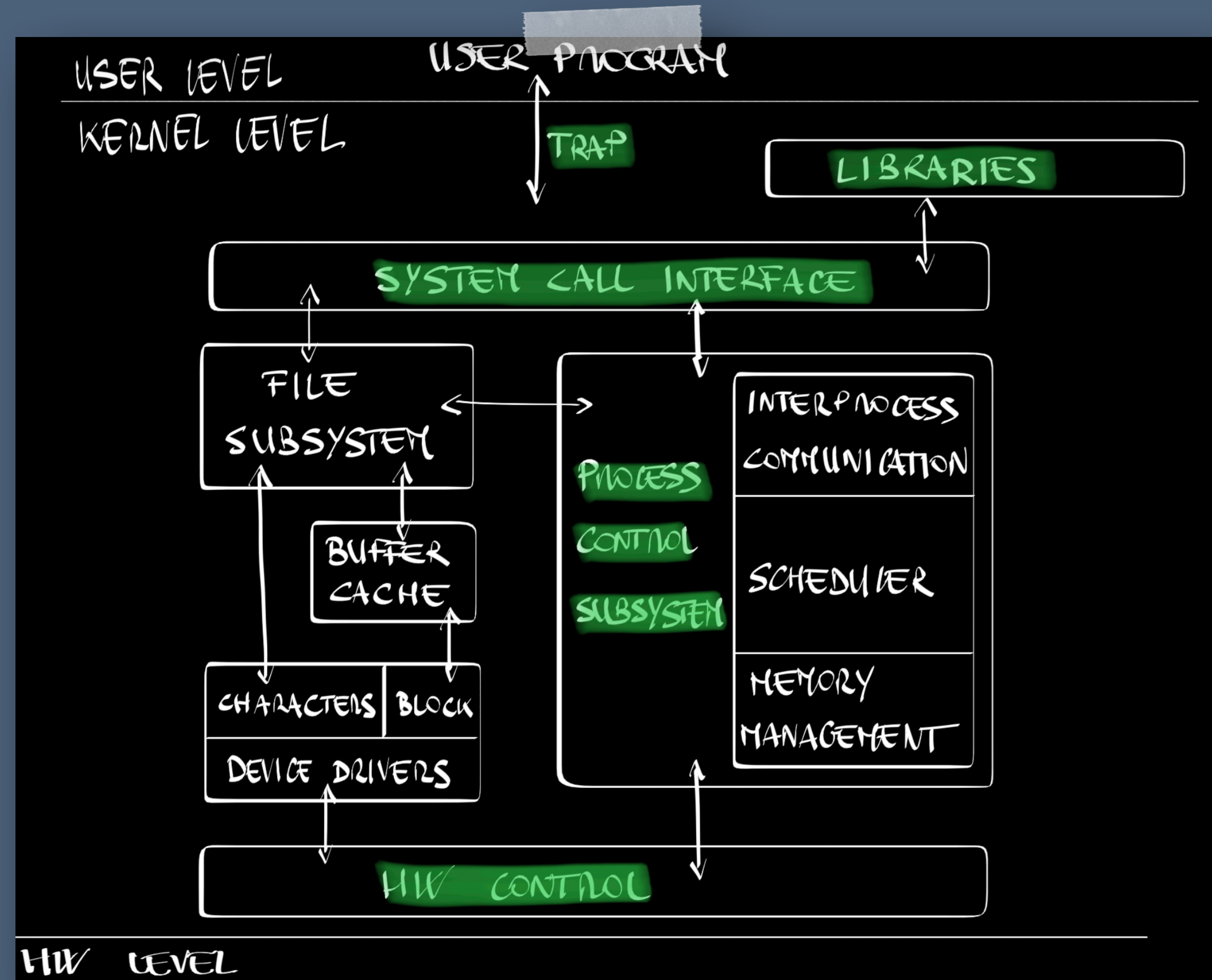
Struttura

System calls

Le system call sono funzioni delle librerie che vengono eseguite direttamente in user mode senza coinvolgere il kernel.

Libraries

Servono per facilitare l'invocazione delle system call



BUG

Le system call sono funzioni delle librerie che vengono eseguite direttamente in user mode senza coinvolgere il kernel.

Correzione:

Vengono invocate dalla user mode e chiamano il SO in kernel mode

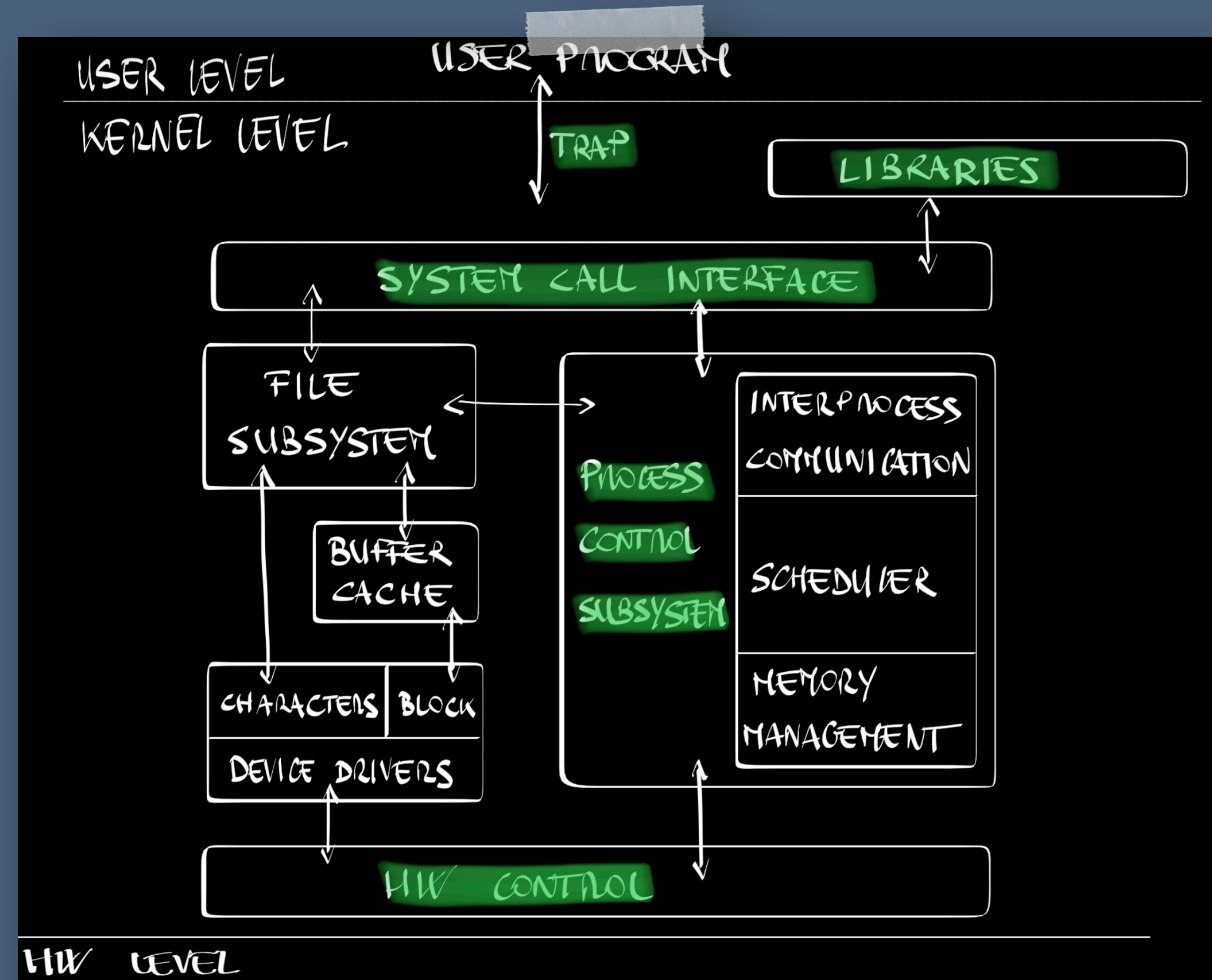


Introduzione ai Sistemi Operativi

Struttura

Trap

Istruzioni macchina usate per implementare le system call.



Cos'è e cosa fa

Introduzione ai Sistemi Operativi

Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Definizione operativa

Un Sistema Operativo è un programma che:

1. Controlla l'esecuzione dei programmi applicativi, fornendo accesso a:
 - Risorse fisiche (CPU, memoria, dispositivi)
 - Risorse logiche (file, pipe, messaggi, semafori)
2. Fornisce un'interfaccia astratta verso l'hardware

È software che controlla il software e fornisce un'interfaccia all'hardware

Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Il SO come macchina virtuale

- Ogni applicativo “vede” la propria macchina virtuale
- La macchina virtuale è più semplice da programmare
- Due visioni:
 1. Gestore di risorse (top-down)
 2. Macchina estesa (bottom-up)

Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Il SO come gestore delle risorse

CPU virtuale → *Multiprogrammazione*

Assegnare una CPU virtuale: la CPU fisica va assegnata "a turno".

Memoria virtuale → *Isolamento e protezione*

La memoria virtuale serve solo ad aumentare la quantità di RAM disponibile.

Assegnare una memoria virtuale: ogni programma deve accedere al proprio testo, ai propri dati, al proprio stack senza interferenze da parte di altri programmi.

BUG

La memoria virtuale serve solo ad aumentare la quantità di RAM disponibile.

Correzione:

La memoria virtuale serve soprattutto a isolamento, protezione e astrazione, non solo ad "avere più RAM"



Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Il SO come gestore delle risorse

Dispositivi virtuali → *Condivisione senza conflitti*

Assegnare ad ogni programma i propri dispositivi virtuali (stampanti, schede di rete,...): i dispositivi vanno condivisi tra i vari programmi evitando interferenze reciproche.

Risorse logiche → *File, semafori, pipe*

Assegnare ad ogni programma le proprie risorse logiche.

Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Il SO come macchina estesa

Il S.O. fornisce API che:

- Nasconde i dettagli dell'hardware
- Favorisce la portabilità dei programmi
- Consente sostituzione di dispositivi senza modificare gli applicativi

Introduzione ai Sistemi Operativi

Cos'è e cosa fa

Obiettivi

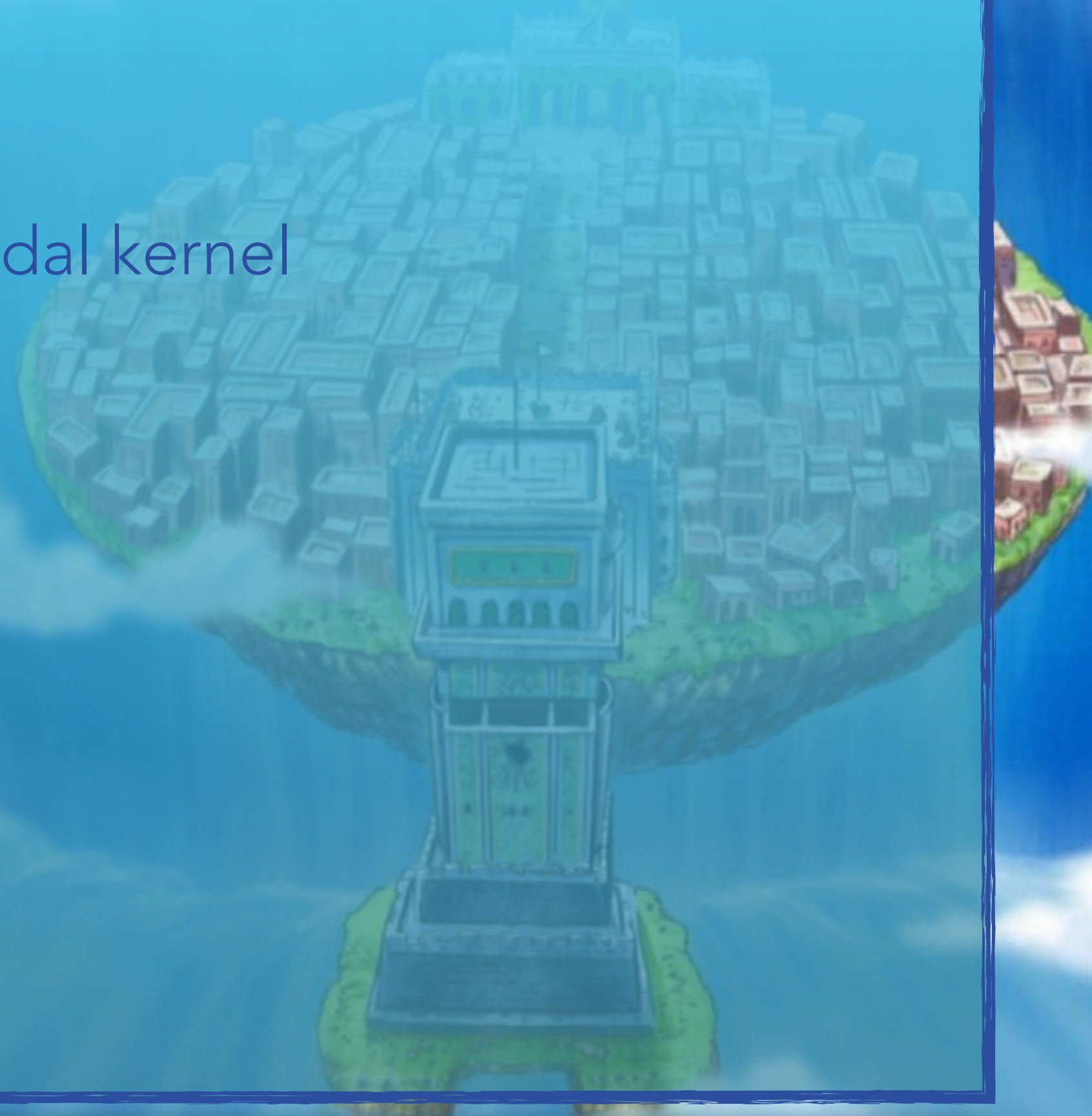
- Uso efficiente del sistema di calcolo
 - Monitoraggio e allocazione risorse
 - Minimizzazione overhead
- Convenienza per l'utente
 - Esecuzione programmi, uso file system
 - Risposte rapide
 - Facilità d'uso (GUI)
 - Strumenti avanzati per esperti

Spesso obiettivi in competizione

ESERCIZIO

Disegna lo schema a livelli di un sistema operativo e indica:

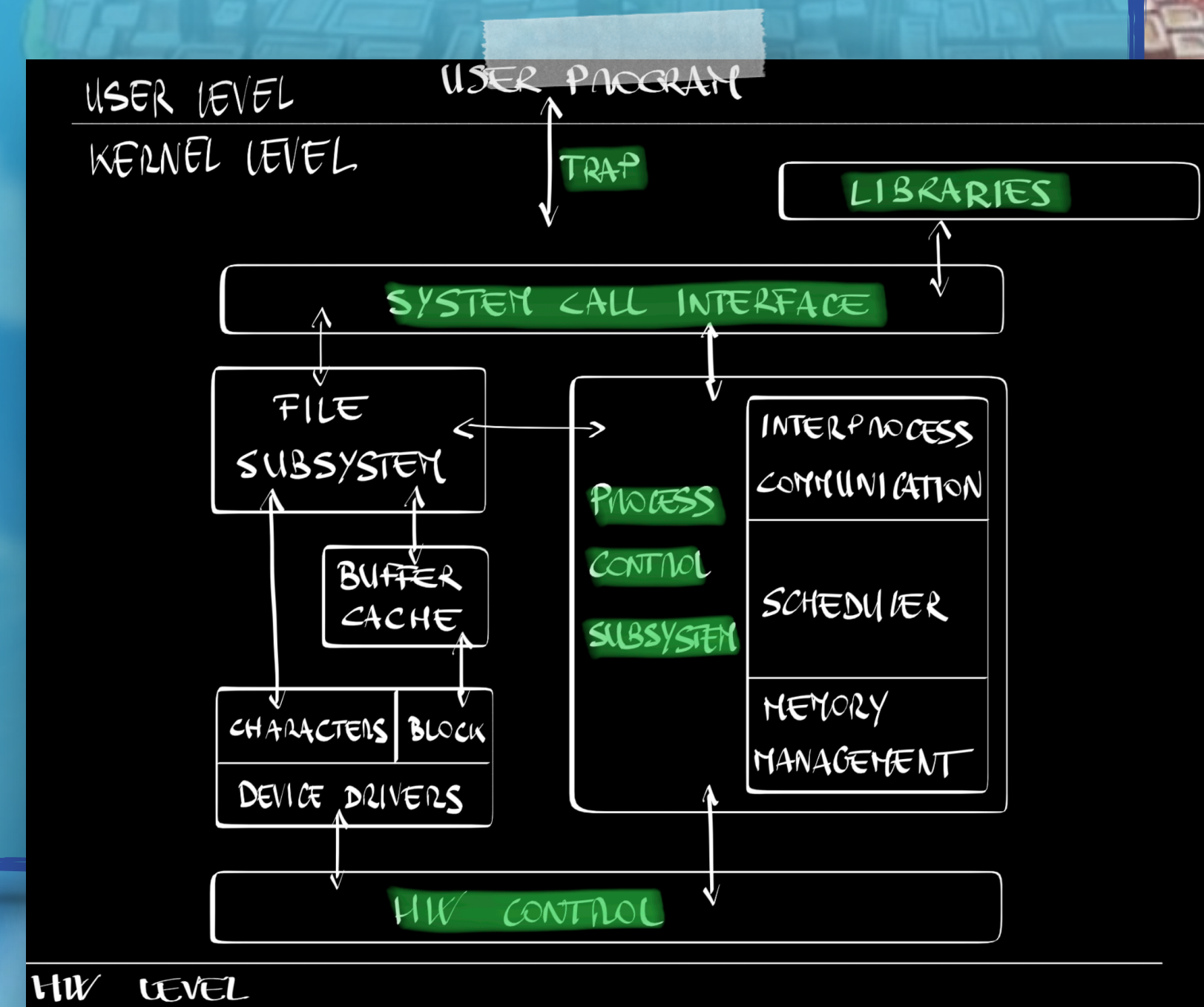
- user level
- kernel level
- system call
- trap
- un esempio di risorsa gestita dal kernel



ESERCIZIO

Disegna lo schema a livelli di un sistema operativo e indica:

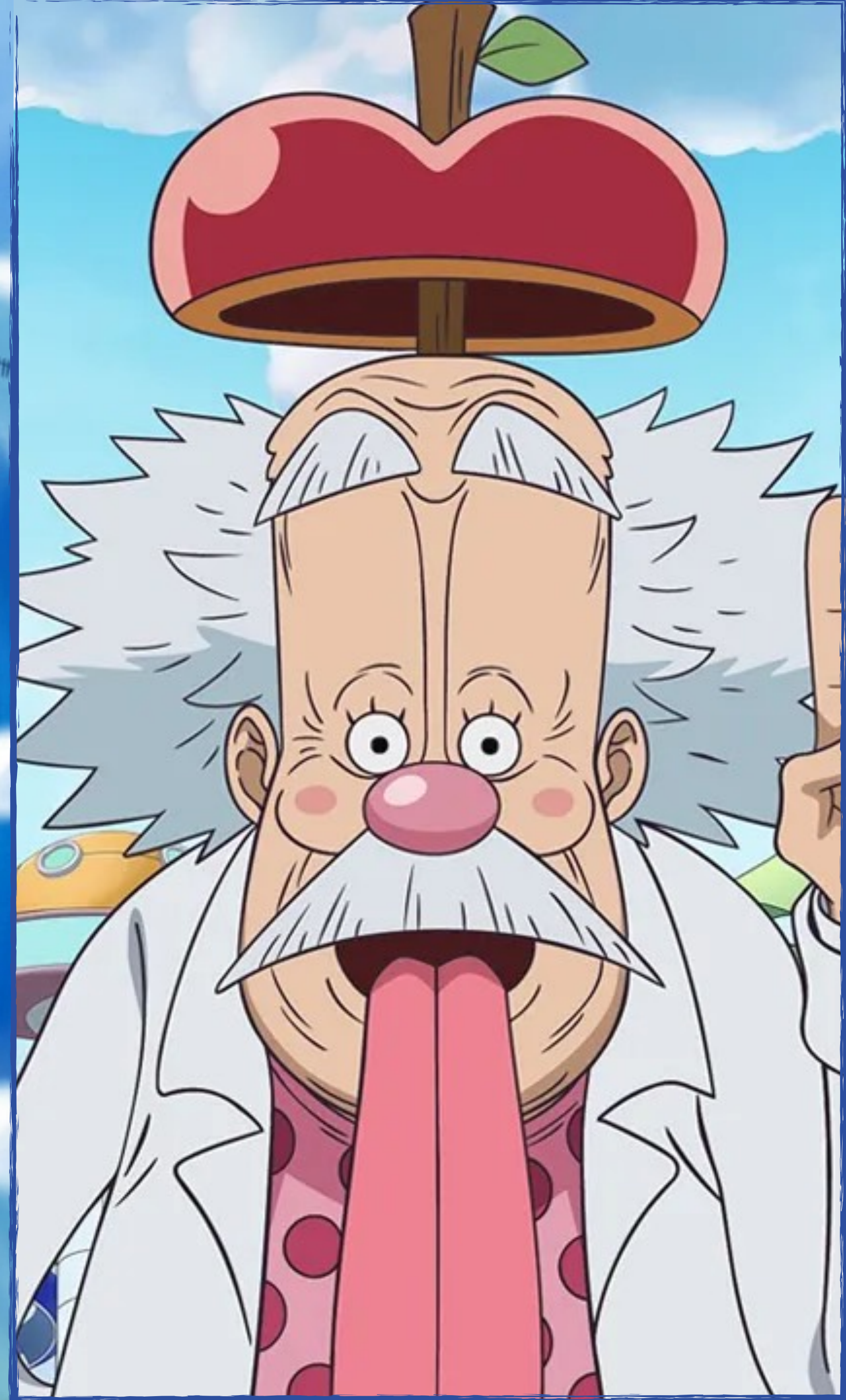
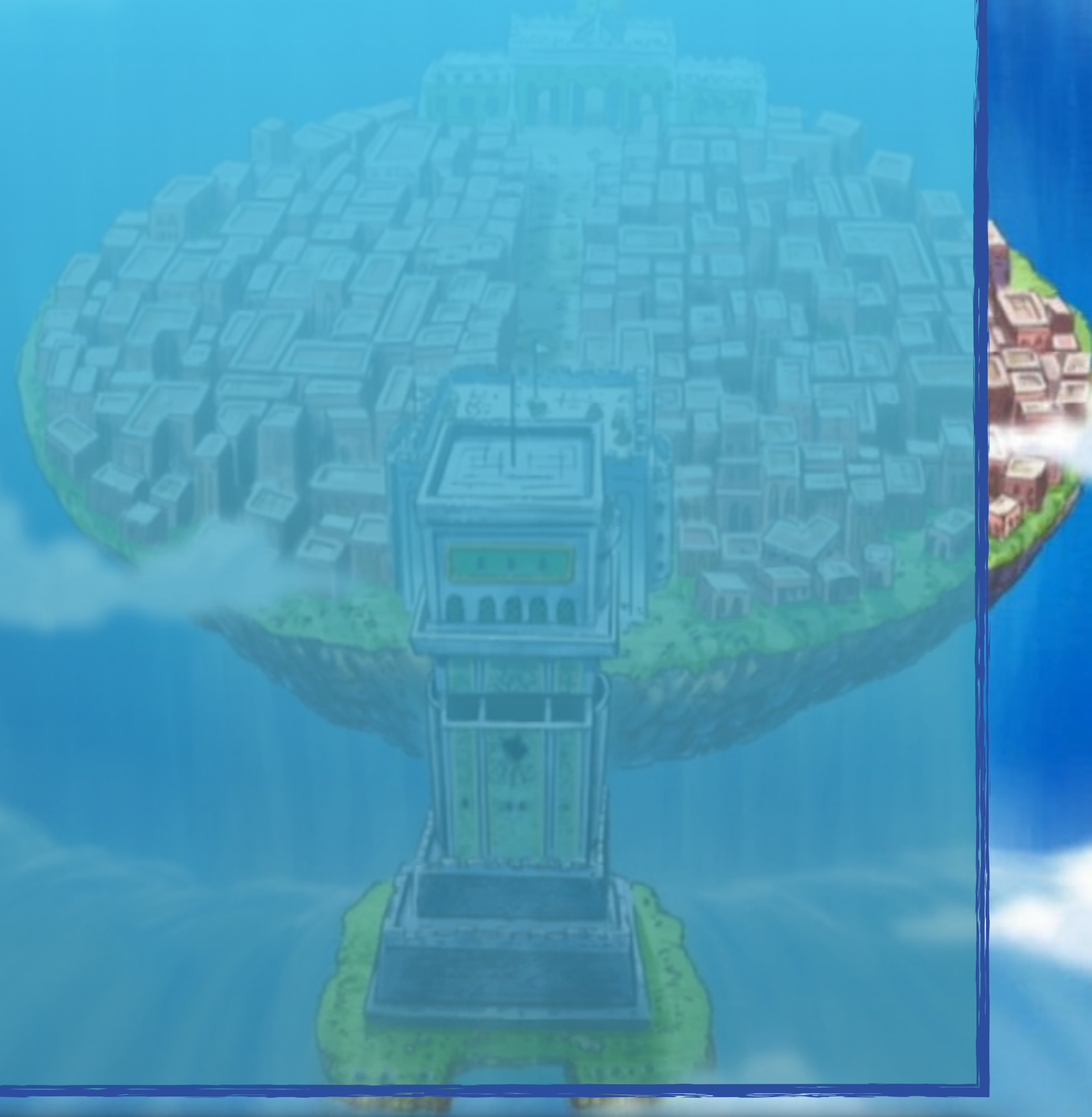
- user level
- kernel level
- system call
- trap
- un esempio di risorsa gestita dal kernel



MEMORIA

"L'astrazione è uno degli strumenti più potenti dell'informatica."

(Edsger W. Dijkstra)



COMPITI

1. Spiega in 10 righe massimo perché il SO viene visto come:
 - gestore di risorse (top-down)
 - macchina estesa (bottom-up)
2. Fai un esempio concreto (Windows, Linux o macOS) di:
 - una system call
 - una risorsa fisica
 - una risorsa logica(indicando quale parte è in user e kernel mode)
3. Rispondi: perché permettere alle applicazioni di eseguire istruzioni privilegiate sarebbe pericoloso?

