

INTRODUCCIÓ

Handler

"Handler" té diferents significats depenent el context:

- **Interrupcions**: És codi (i.o. funcions) que s'executen en resposta a un event.
- **Recursos**: És un identificador (Número o struct) que S.O. facilita per interactuar amb aquest.

Tipus de Handlers

IRS

★ **Interrupt Handler** (Interrupt Service Routine): Handler molt específic que normalment gestiona interrupcions de HW o SW. (Codi)

★ **Resource Handler**: Handler que permet interactuar còmodament amb el recurs del S.O.
Per exemple: Amb un fitxer, Socket, Clipboard, ...

Interrupt Vector Table (IVT)

És una "data structure" que associa Interrupt Handler ↔ Interrupt Request.
Cada entrada (f.b.) és un Interrupt Vector.

Un Interrupt Vector conté la direcció de memòria del Interrupt Handler corresponent.
Aquest és un vector pq. pot ser que en futur les direccions de mem siguin més grans.

Llibreria

Conjunt de codi preexistent desenvolupat per agilitzar desenvolupant o tanques.

Llibreria del Sistema

Permet la interacció directa amb el kernel i gestionar recursos del sistema.

Tot i que interactui amb kernel està en "la banda" de l'usuari pq. té funcions ben definides i que no són "perilloses" com per exemple: `open()`, `fork()`, `exec()`, ...

⚠ **RECORDA**: Llibreria conté els "s.o." i a l'hora de compilar programes s'enllacen (linker).

Exemple: Tenim un programa "p1.c" que fa ús de "`printf()`". Gràcies a la llibreria de C (libc) sabem que fa la funció de "`printf`". Probablement aquesta funció fa ús de "`write()`" que és una funció de la llibreria del sistema. Internament aquesta funció del sistema saltarà a "`sys_write`" on ja serà el codi kernel que s'executa.

⚠ **IMPO**: Hi ha funcions de C que es diuen igual que les sys call però no són el mateix.

Interrupcions de Relletge

Mecanisme del sistema operatiu que permet assegurar-se que cap process monopolitzi l'ús de la CPU. S.O. programa interrupció cada X temps atquant es provocat pel temporitzador. El S.O. evalua (cada vegada que hi ha una interrupció de relletge) si el process ha consumit el seu temps assignat en la CPU. Si això passa, el planificador decideix. # En futur veuram diferents implementacions i millores explicacions.

"Cultura General"

"kernel" == "Sistema Operatiu".

| Si volem més detall de l'error

if (return == -1) "Hi ha hagut error";

| hem de comparar el número de la

else "No hi ha hagut cap";

| variable global "errno".