

Fonaments dels Sistemes Operatius (FSO)

Departament d'Informàtica de Sistemes i Computadores (DISCA)

Universitat Politècnica de València

Bloc Temàtic 2: Gestió de Processos

Unitat Temàtica 3

Procés:

Concepte i implementació

fSO

DISCA



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

- **Objectius:**

- Definir el concepte de **procés**
- Analitzar les diferències entre execució **seqüencial i concurrent**
- Definir els **estats d'un procés** així com les causes de les transicions entre ells
- Estudiar **l'estructura bàsica** per a donar suport als processos en el SO
- Analitzar el mecanisme del **canvi de context**

- **Bibliografia:**

- “Fundamentos de sistemas operativos” Silberschatz 7ª Ed
- “Sistemas operativos: una visión aplicada” Carretero 2º Ed

- **Conceptes previs**
- Fitxers executables
- Concepte de procés
- Estats d'un procés
- Implementació de processos: el PCB
- Taula de crides per a processos i senyals
- Exercicis proposats

- Per a poder entendre el concept de procés, es importante tindre clars alguns termes. En concret les diferències entre:
 - Programa/Procés
 - Execució seqüencial / Execució concurrent

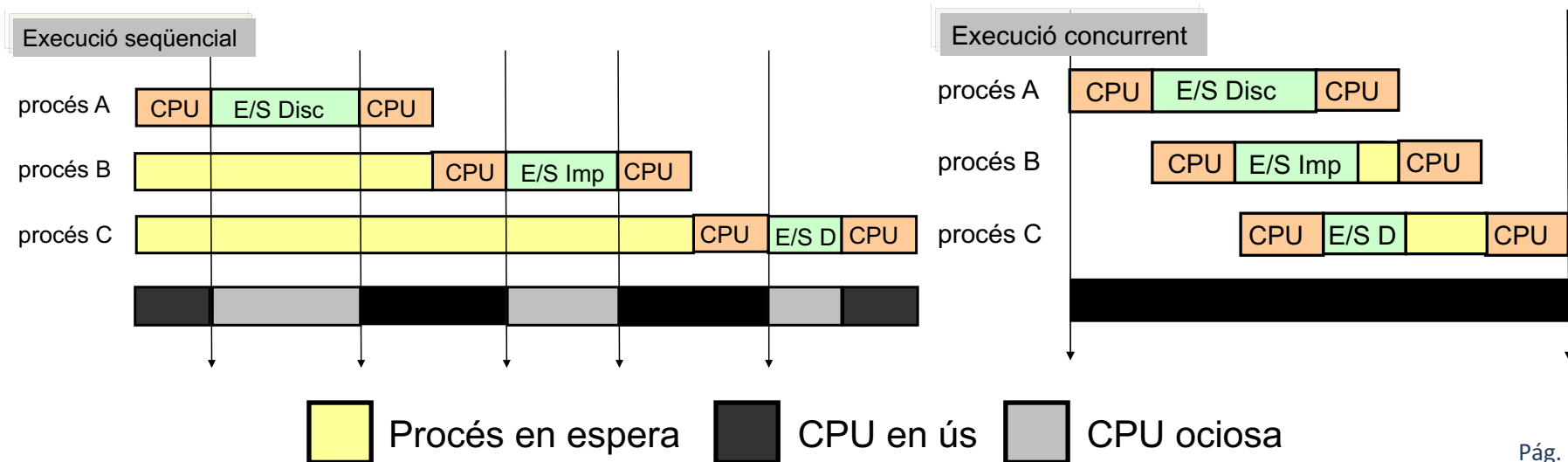
Programa vs. Procés

- **Programa:**
 - **Fitxer** executable resultat d'un procés de compilació i enllaçat correcte.
 - **Entitat passiva**, no canvia amb el temps.
- **Procés:** instància d'un programa en execució:
 - Unitat de treball del sistema operatiu
 - Consumidor de recursos
 - Qualsevol tasca que execute un computador ha de ser un procés
 - **Abstracció del sistema operatiu** que modela les activitats que apareixen en cada moment en el computador

Un procés és una **entitat activa** que pateix canvis mentre existeix

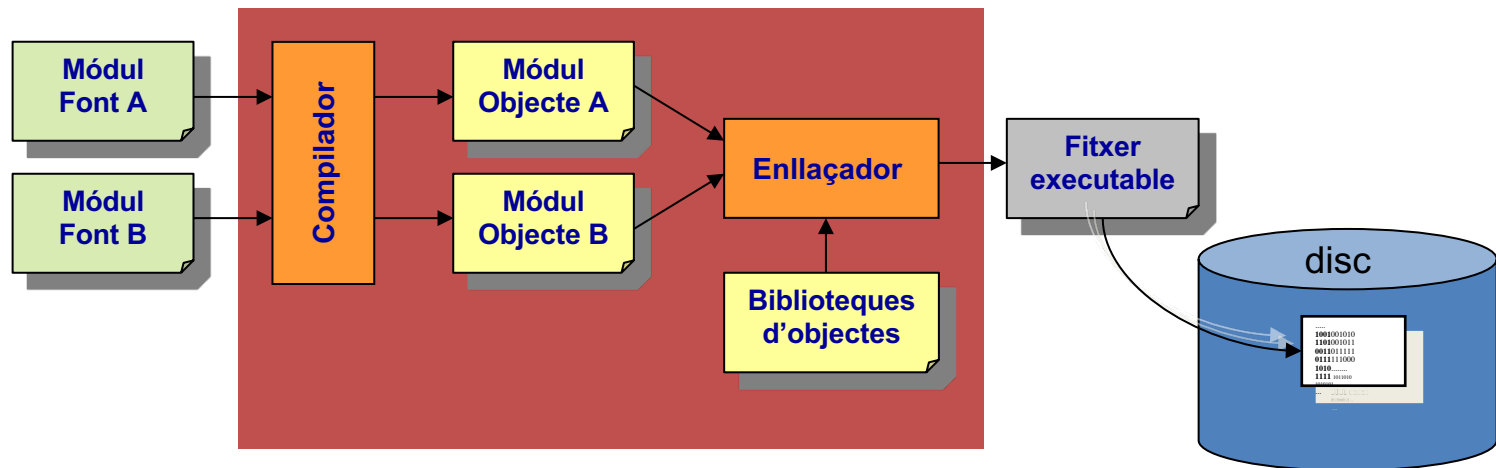
Execució seqüencial vs. Execució concurrent

- **Execució seqüencial:** quan la CPU és utilitzada per un sol procés des de que s'inicia la seua execució fins que acaba.
- **Execució concurrent:** quan la CPU es utilitzada per diversos processos alternant-ne l'ús durant la seua vida (*multiprogramació*).
 - El benefici més directe de l'execució concurrent és l'increment d'ús de la CPU, donat que un procés no es troba constantement demanant CPU si no que alterna les seues demandes amb ràfegues d'E/S.

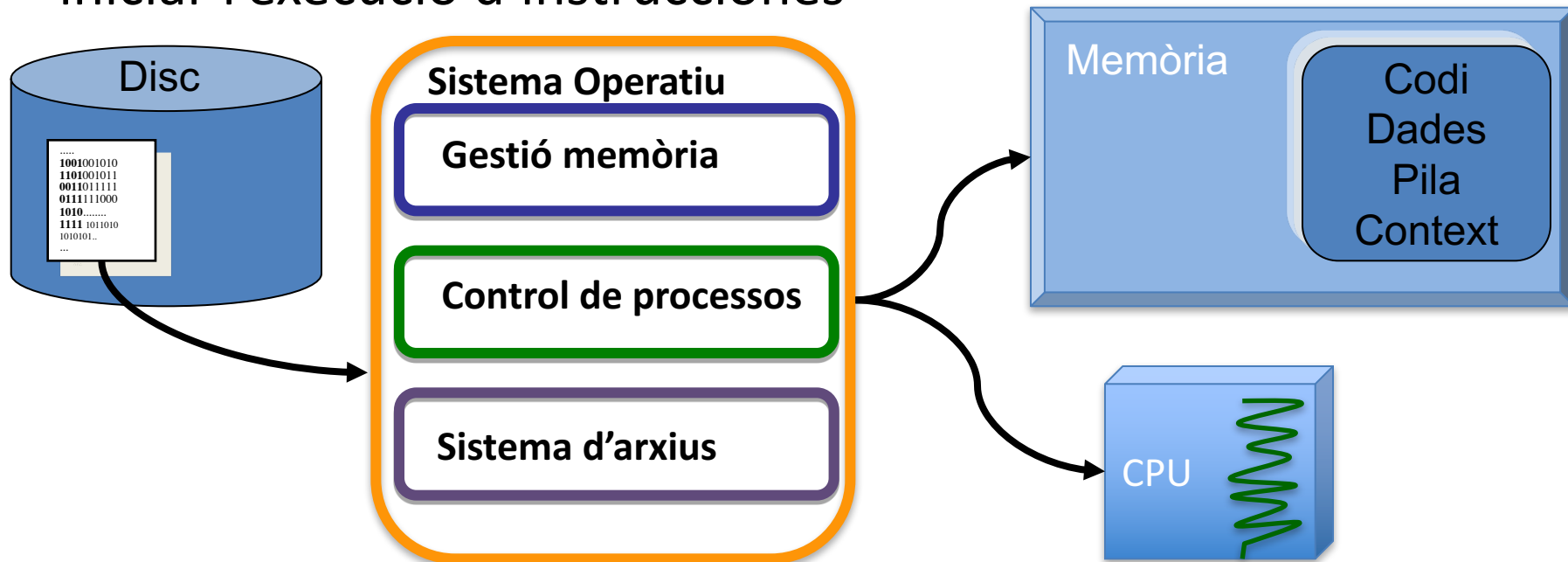


- Conceptes previs
- **Fitxers executables**
- Concepte de procés
- Estats d'un procés
- Implementació de processos: el PCB
- Taula de crides per a processos i senyals
- Exercicis proposats

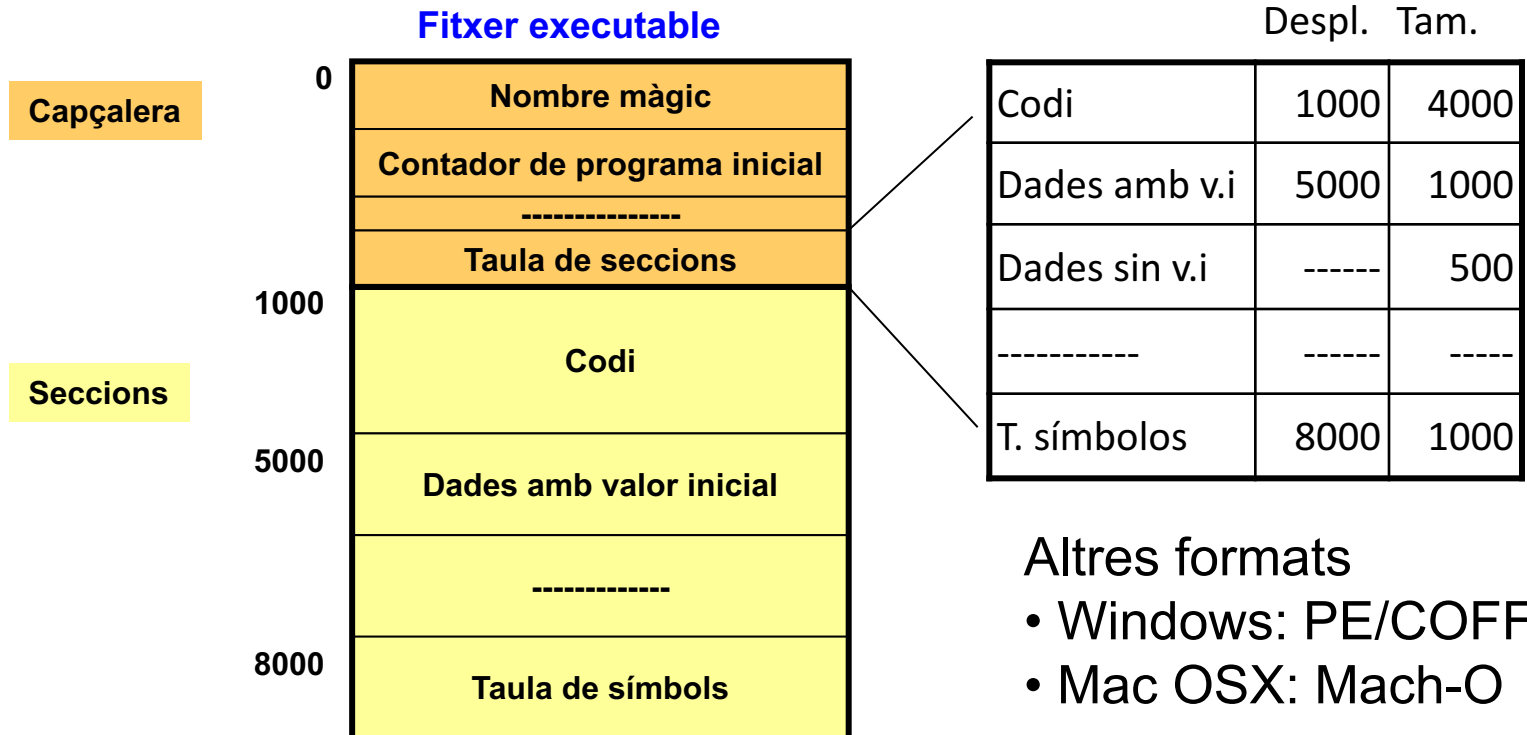
- Per a llançar a execució un procés es necessari partir d'un **fitxer executable**, es a dir, un fitxer que continga codi executable
- Per a obtindre aquest fitxer s'han de seguir els següents passos:
 1. Disposar d'un fitxer de text amb el programa escrit en un llenguatge d'alt nivell, que denominarem **codi font**
 2. Compilar el fitxer amb el **codi font** per a obtindre un **codi objecte**
 3. El codi objecte obtés **s'enllaça amb** el codi de **biblioteques** del sistema o altres d'usuari. El resultat final és un **fitxer executable**
 - Les biblioteques ens donen la possibilitat d'incorporar codi extern



- Un executable és un fitxer l'estructura del qual és ben coneguda pel sistema operatiu, ja que conté:
 - El codi a executar
 - Les dades inicialitzades
 - Les funcions de biblioteca
- Amb tota aquesta informació el SO pot assignar els recursos necessaris per a la seua execució. Carregar dades, codi etc. i iniciar l'execució d'instruccions



- Com a exemple de format de fitxer executable tenim els **fitxers ELF b(Executable and Linking Format)**
 - És un format de fitxer per a executables, codi objecte, biblioteques i volcats de memòria.
 - Molt extès en Unix, Linux, Solaris i BSD.
 - Extensions: .o, .so, .elf , .prx, .exe, .dl



- Conceptes previs
- Fitxers executables
- **Concepte de procés**
- Estats d'un procés
- Implementació de processos: el PCB
- Taula de crides per a processos i senyals
- Exercicis proposats

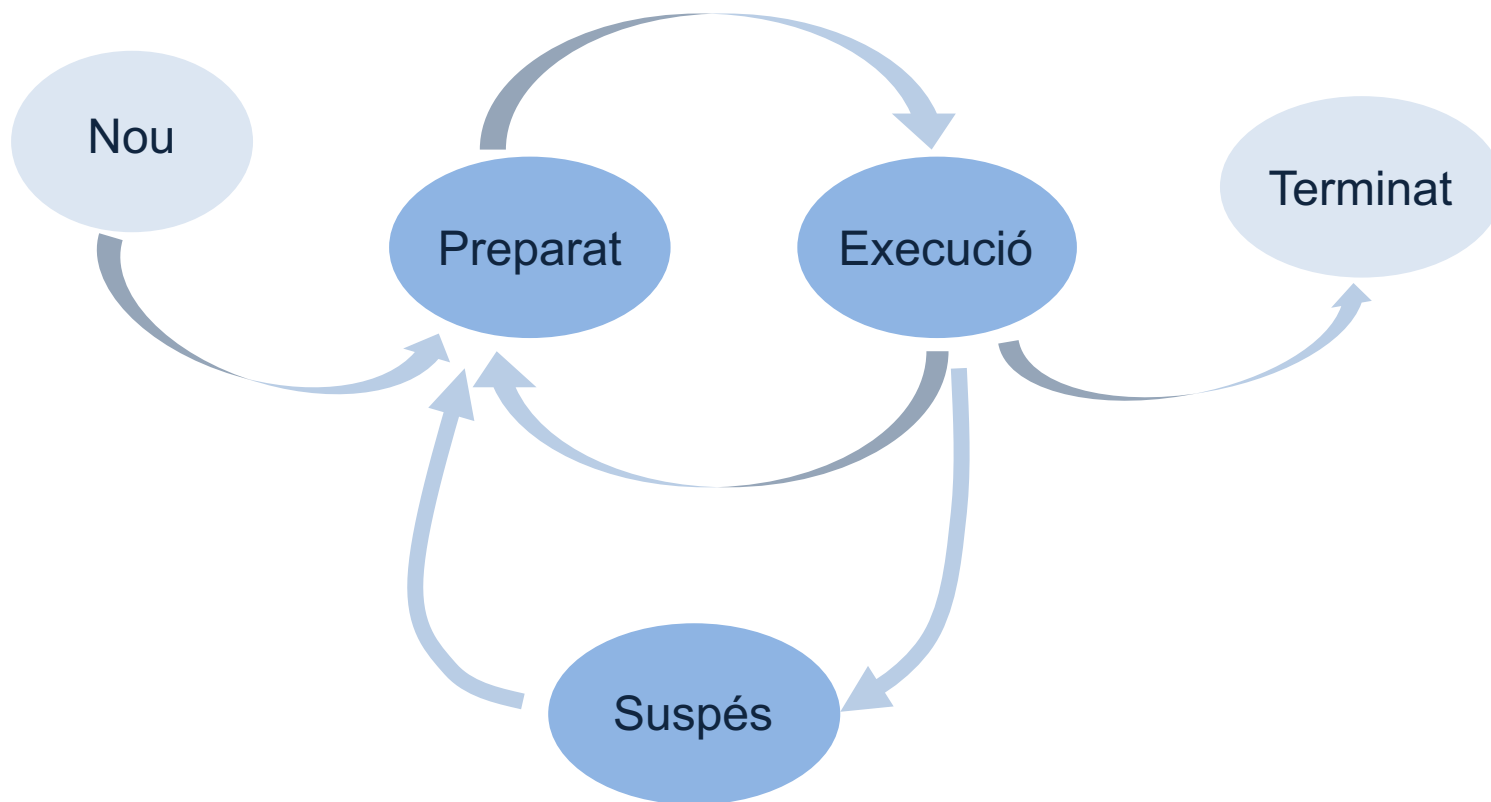
- Per a definir el concepte de procés i arribar a la seua implementació, és necessari definir tres aspectes fonamentals:
 - **Atributs** o característiques que ajuden a definir-los i a gestionar-los dins del sistema
 - **Comportament**, al ser un ens actiu, pot passar per diferents estats i és necessari definir tant els estats com les transicions entre ells
 - **Operacions** que es poden realitzar sobre ells

Atributs d'un procés

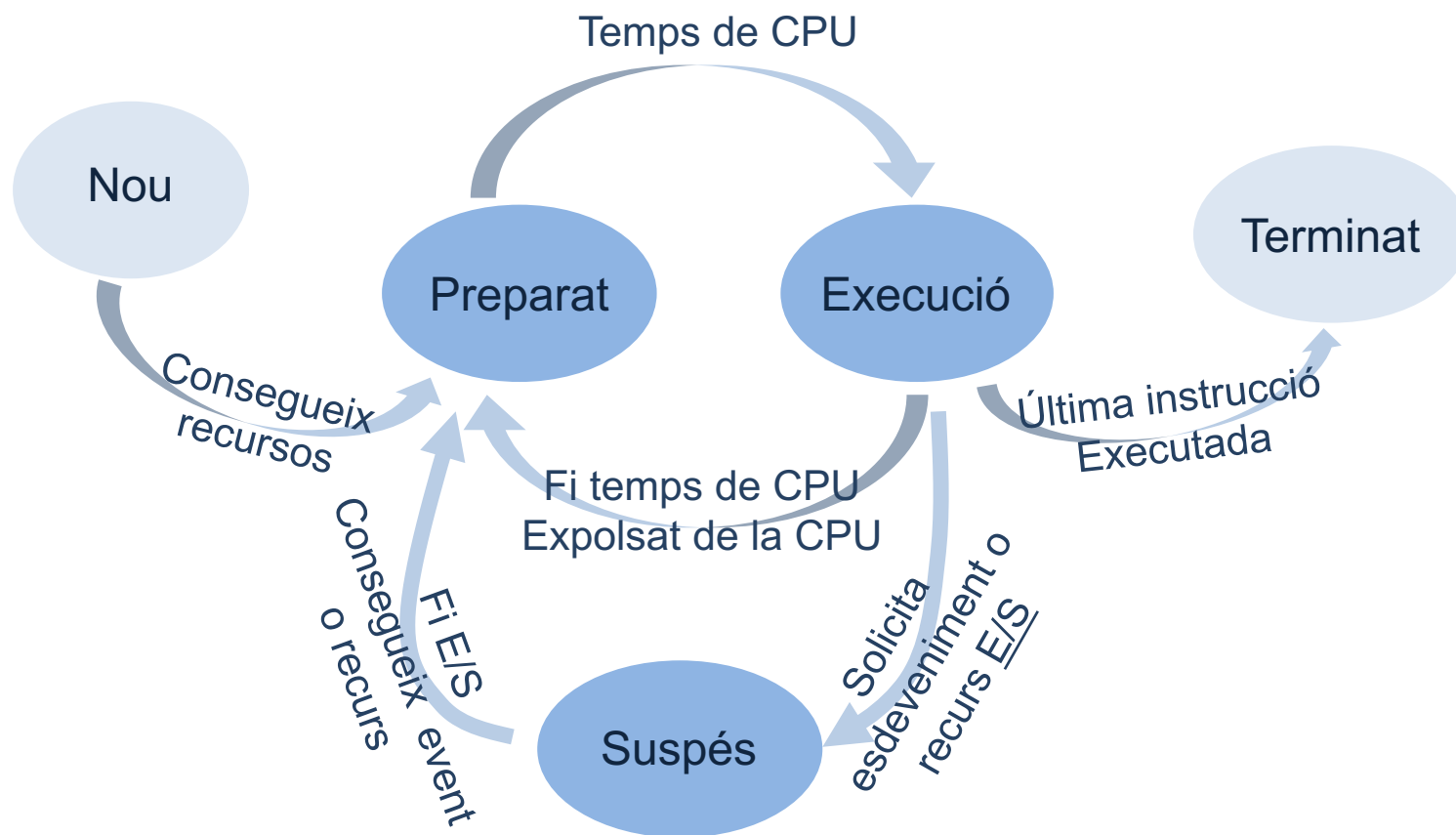
- Els atributs d'un procés són aquelles propietats, recursos o característiques pròpies del procés que manté el SO per a poder gestionar-los
- A pesar de que cada sistema operatiu manté un conjunt d'atributs que poden variar, els més típics solen ser:
 - **Atributs d'identitat:**
 - Identificador del procés
 - **Atributs d'entorn d'execució:**
 - Directori actual
 - Descriptors de fitxers oberts
 - **Atributs d'estat:**
 - Estat del procés
 - Context màquina (comptador de programa, punter pila, registres us general)
 - Etc.
 - **Atributs de memòria:**
 - Àrea de codi, àrea dades, àrea pila
 - **Atributs de planificació:**
 - Temps de consum de processador
 - Prioritat
 - Monitorització

- Conceptes previs
- Fitxers executables
- Concepte de procés
- **Estats d'un procés**
- Implementació de processos: el PCB
- Taula de crides per a processos i senyals
- Exercicis proposats

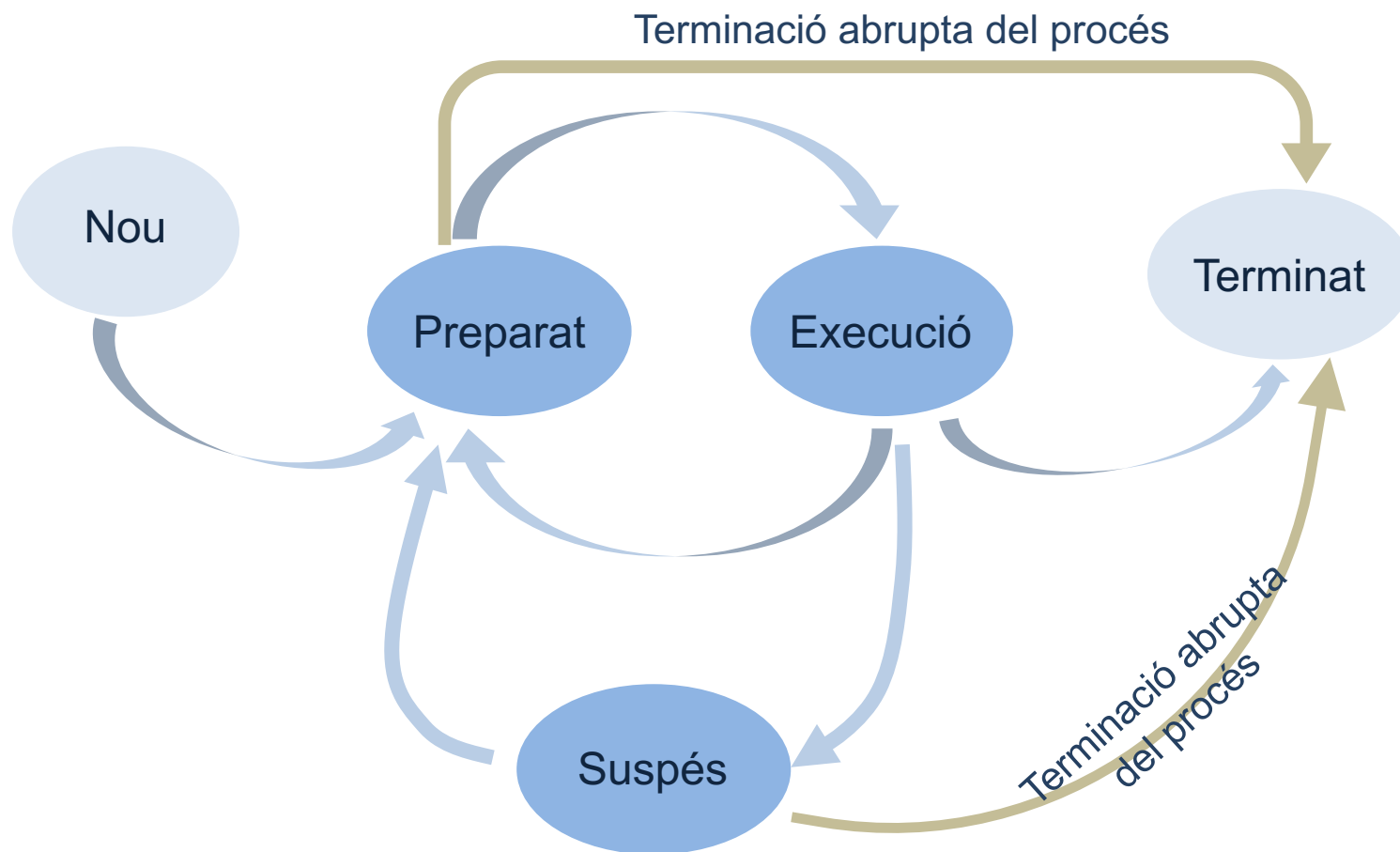
- Els **processos** són **entitats actives** i, per tant, passen per **diferents estats** al llarg de la seua vida

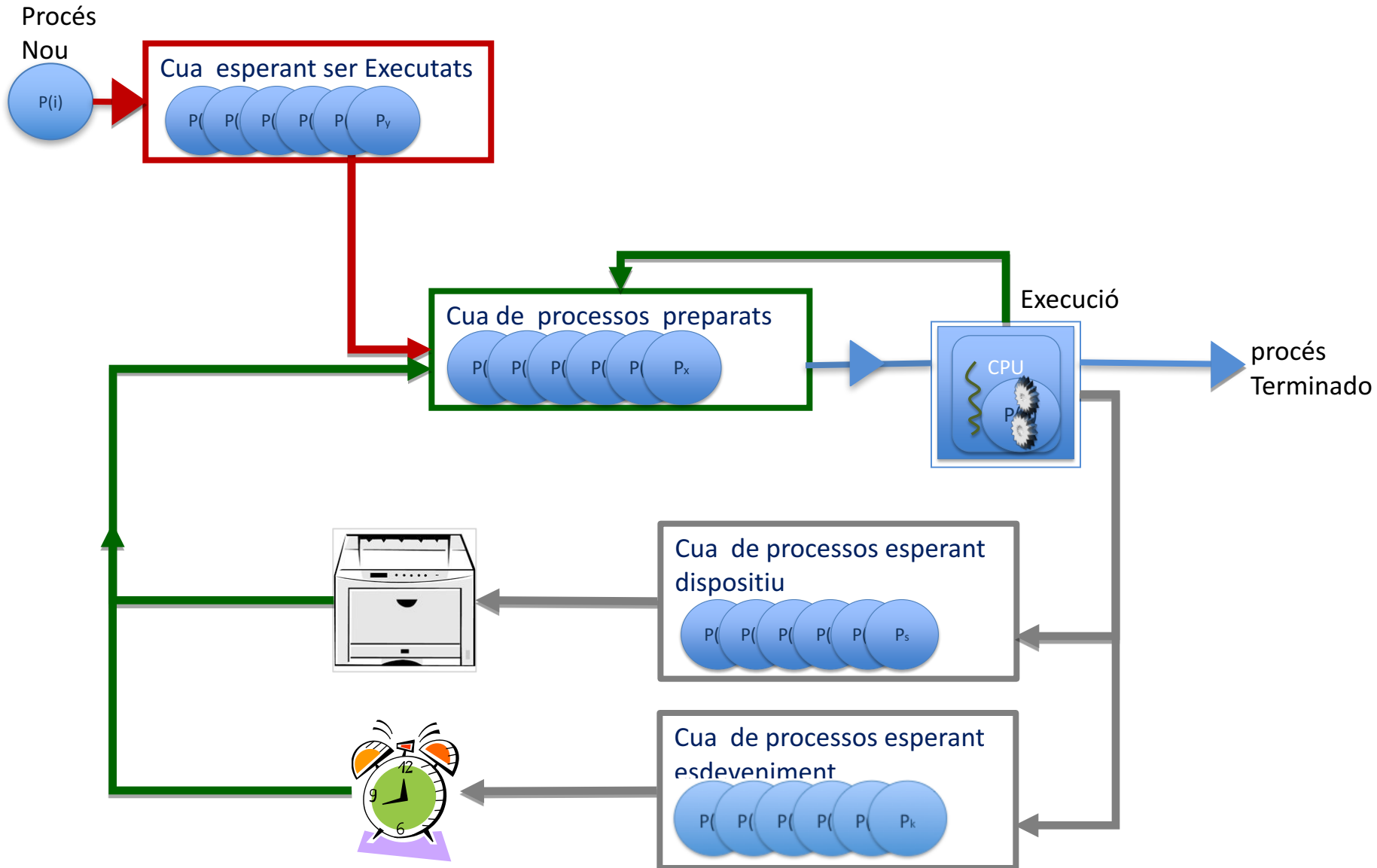


- Transicions entre estats d'un procés



- **Transicions causades per situacions anòmales**



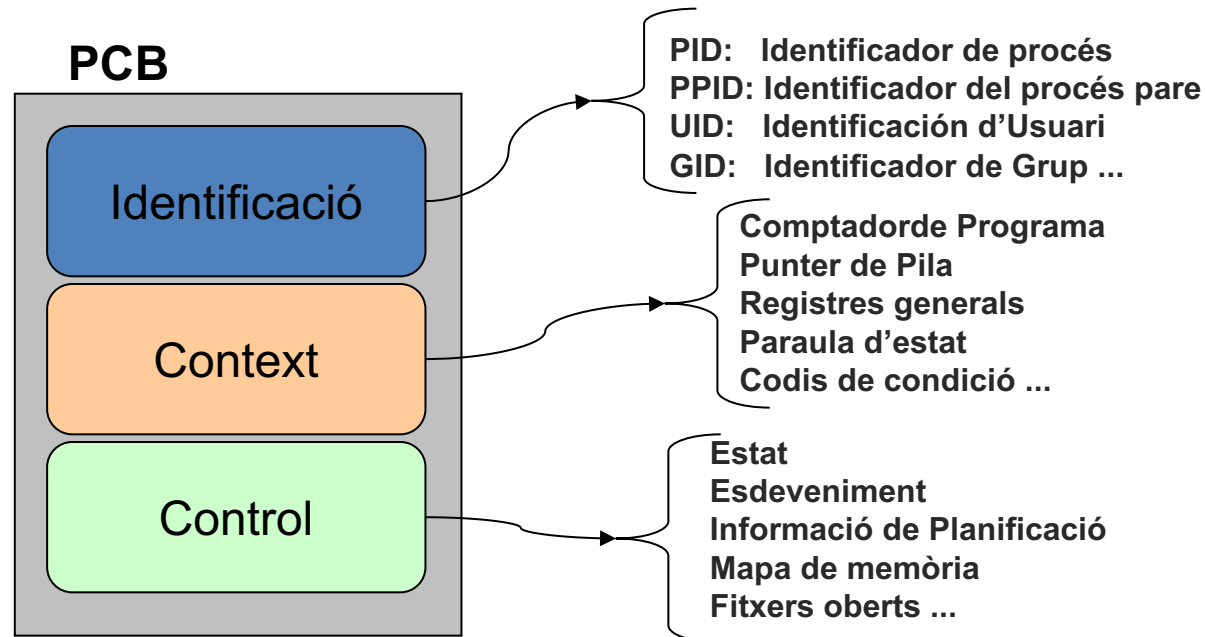


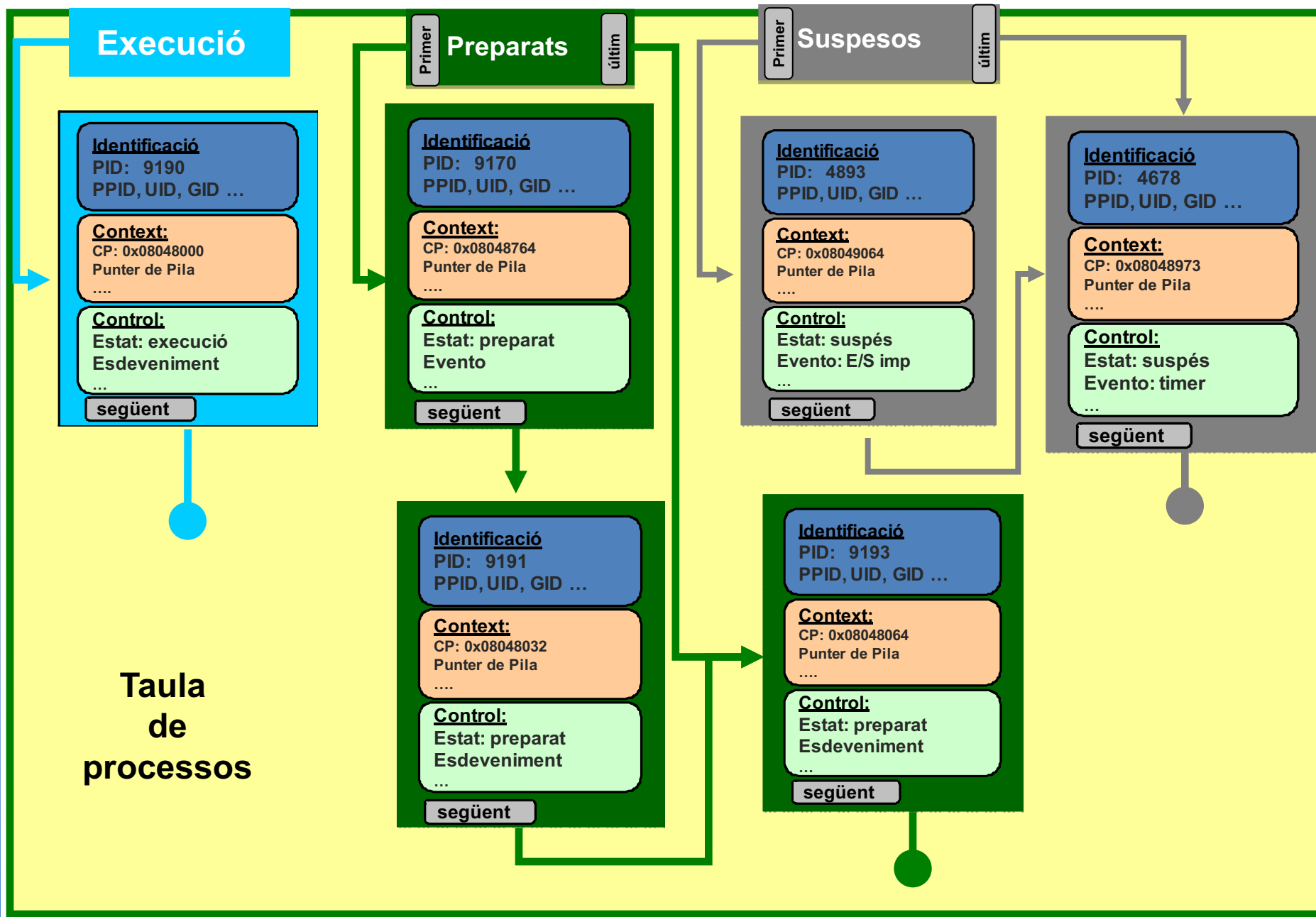
- **Operacions sobre processos**
 - Com passava amb els atributs, el nombre i tipus de les **operacions** que es poden fer sobre processos **varia en funció del sistema** operatiu.
 - No obstant, sempre podem trobar operacions de:
 - Creació
 - Comunicació
 - Espera
 - Accés a recursos
 - Finalització

- Conceptes previs
- Fitxers executables
- Concepte de procés
- Estats d'un procés
- **Implementació de processos: el PCB**
- Taula de crides per a processos i senyals
- Exercicis proposats

- **PCB (Process Context Block)**

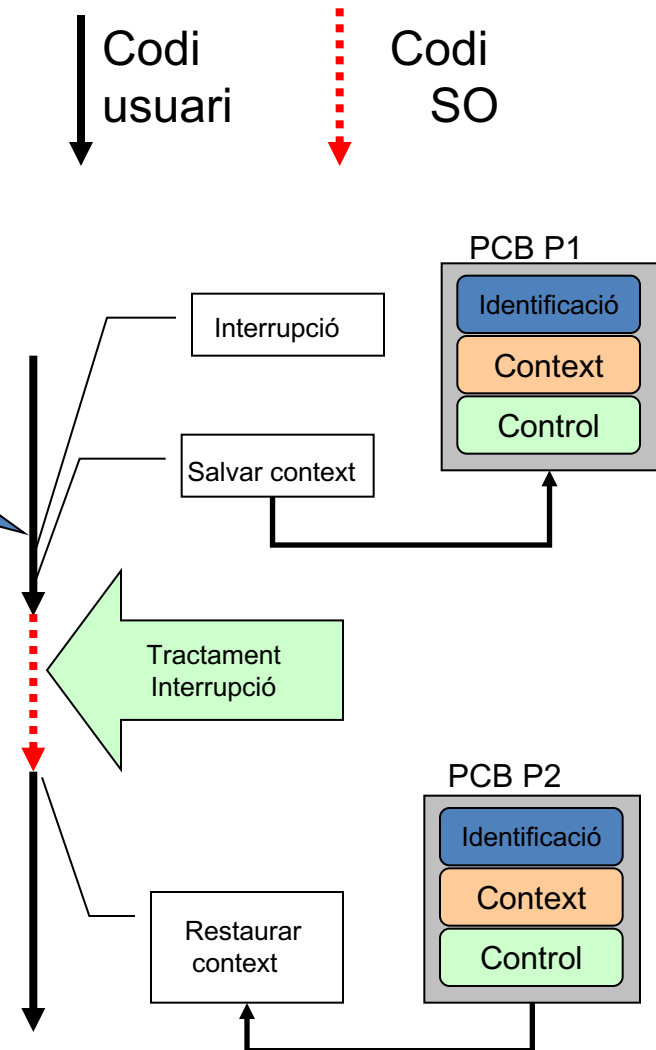
- Un PCB (Process Context Block) és l'**estructura de dades** sobre la que el SO sustenta el concepte (abstracció) de procés
 - Un SO també és un programa, es basa en l'ús d'**algoritmes i estructures de dades**
- Manté informació relevant del procés, la qual canvia durant la vida del mateix.
- Cada sistema operatiu té la seua pròpia estructura.





• Canvi de context

- Mecanisme que permet a un SO detindre la seqüència actual d'execució d'un procés per a iniciar o reprendre l'execució d'altre procés
 - Aquest mecanisme s'activa mitjançant una interrupció (p. Ex. Interrupció de rellotge)
- ¿Què s'hi fa?
 - D'estalvia la informació rellevant de l'estat del procés en execució
 - S'executa el codi del SO associat a la interrupció
 - Es restaura en la CPU l'estat d'un procés per a que continúe la seua execució



- Conceptes previs
- Fitxers executables
- Concepte de procés
- Estats d'un procés
- Implementació de processos: el PCB
- **Taula de crides per a processos i senyals**
- Exercicis proposats

	Processos
fork	Creació d'un procés fill
exit	Acabament del procés en execució
wait	Espera l'acabament d'un procés
exec	Canvia la imatge de memòria per la d'un executable (executa programa)
getpid	Obté atributs d'un procés
setsid	Modifica atributs d'un procés

	Senyals
kill	Enviar senyals
alarm	Generar una alarma (senyal de rellotge)
sigemptyset	Iniciar una màscara per a que no tinga senyals seleccionades
sigfillset	Iniciar una màscara per a que continga tots els senyals
sigaddset	Afegir un senyal concret a un conjunt de senyals
sigdelset	Esborrar un senyal concret d'un conjunt de senyals
sigismember	Consultar si un senyal concret pertany a un conjunt de senyals
sigprocmask	Examinar /Modificar /Establir una màscara de senyals
sigaction	Capturar/Manejar un senyal
sigsuspend	Esperar la captura de senyals

- Conceptes previs
- Fitxers executables
- Concepte de procés
- Estats d'un procés
- Implementació de processos: el PCB
- Taula de crides per a processos i senyals
- **Exercicis proposats**

Exercici UT03.1_Donada la següent llista d'accions, digueu per a cadascuna d'elles si és el codi del sistema operatiu (SO) i/o el codi de l'interpret d'ordres (IO) el responsable de dur-les a terme. Indiqueu-lo marcant amb una creu .

SO	IO	
		Llegir una línia d'ordres i interpretar-la
		Programar un controlador de dispositiu
		Proporcionar una interfície de crides al sistema
		Seleccionar un procés per a assignar-li CPU
		Invocar una crida al sistema
		Proporcionar una interfície còmoda d'usuari

Exercici UT03.2 Quin seria l'estat (nou, preparat, execució, suspès, acabat) en què es trobarien cadascun dels processos següents?

Procés		Estat
P1	La CPU està executant instruccions de P1	
P2	P2 ha sol·licitat un accés a disc, però el disc està següent accedit per P3	
P3	P3 està accedint al disc	
P4	P4 és un procés que correspon a un usuari que finalitza tots els seus treballs en un terminal i es desconnecta	
P5	Al procés P5 se li ha assignat un identificador de processos i només s'han construït les taules necessàries per a gestionar-lo	
P6	Les taules necessàries per a la gestió de P6, així com la seua imatge de memòria, han sigut carregades en la memòria	

Exercici UT03.3 Donat el següent codi t línia d'ordres:

```
$ cat f1 f2 f3 | grep comença | wc -l >traça
```

Indiqueu:

- a) Quants processos es crearien durant la seua execució per un sistema UNIX
- b) Quins arxius d'E/S duu associada cada ordre?

- Donat el següent codi:

```
#include <stdio.h>
#include <sys/types.h>

int main(void)
{
    pid_t pid;
    int i;

    for (i=0; i<2; i++)
        pid=fork();
    return 0;
}
```

Indiqueu:

- Quants processos es crearien durant la seua execució?