

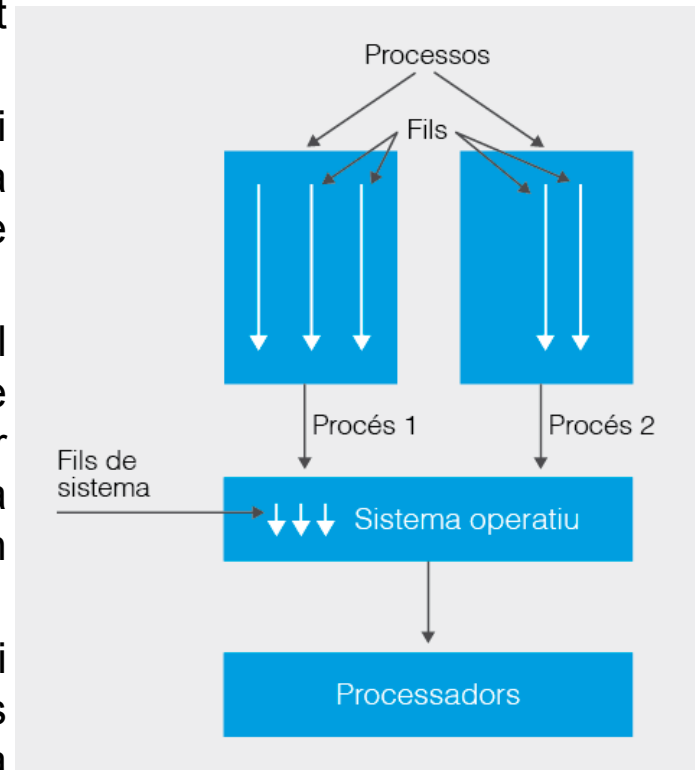
1. Introducció als processos

1. Definició
2. Planificació de processos
3. Estats d'un procés
4. Estats de transició
5. Bloc de Control de Procés
6. Canvi de context
7. Avantatges de la multitasca
8. Criteris de planificació
9. Algorismes de planificació

1. Definició

- Un procés és un programa en execució, és a dir, és una entitat activa.
- Els programes per contra són entitats passives, constituïdes per fitxers que contenen codi font escrit en algun llenguatge de programació.
- Un procés necessita estar carregat en memòria i disposar de determinats recursos (CPU, memòria, arxius, dispositius d'E/S) per tal de dur a terme la seva tasca.
- El component principal d'un SO és el que s'encarrega de la gestió de processos.

- Cada fil es processa de forma independent malgrat que pertanyi o no a un mateix procés.
- L'única diferència entre el tractament de fils i processos la trobem a nivell de memòria. Per a fils d'un mateix procés el sistema operatiu ha de mantenir les mateixes dades en memòria.
- Per a fils de diferents processos en canvi cal disposar de dades independents. En el cas que sigui necessari que un mateix processador alterni l'execució de diversos processos, caldrà restaurar la memòria específica del procés en cada canvi.
- D'això s'anomena també canvi de context. Si l'alternança de processament es fa entre fils d'un mateix procés, no caldrà restaurar la memòria i per tant el procés serà molt més eficient.



2. Planificació de processos

- Un procés perd la CPU degut a:
 - Realitza una crida al sistema que el bloqueja.
 - Rep una interrupció.
 - El procés acaba.
- Planificador: és la part del SO encarregada de seleccionar el següent procés que passarà a l'estat d'execució.
- Algorisme planificador: l'algorisme que selecciona el procés.
- Procés nul: procés que el SO posa en execució quan no existeix cap altre procés en el sistema que pugui estar en execució. El processador sempre està en execució.



Institut Poblenou

Consorci d'Educació
de Barcelona

Generalitat de Catalunya
Ajuntament de Barcelona

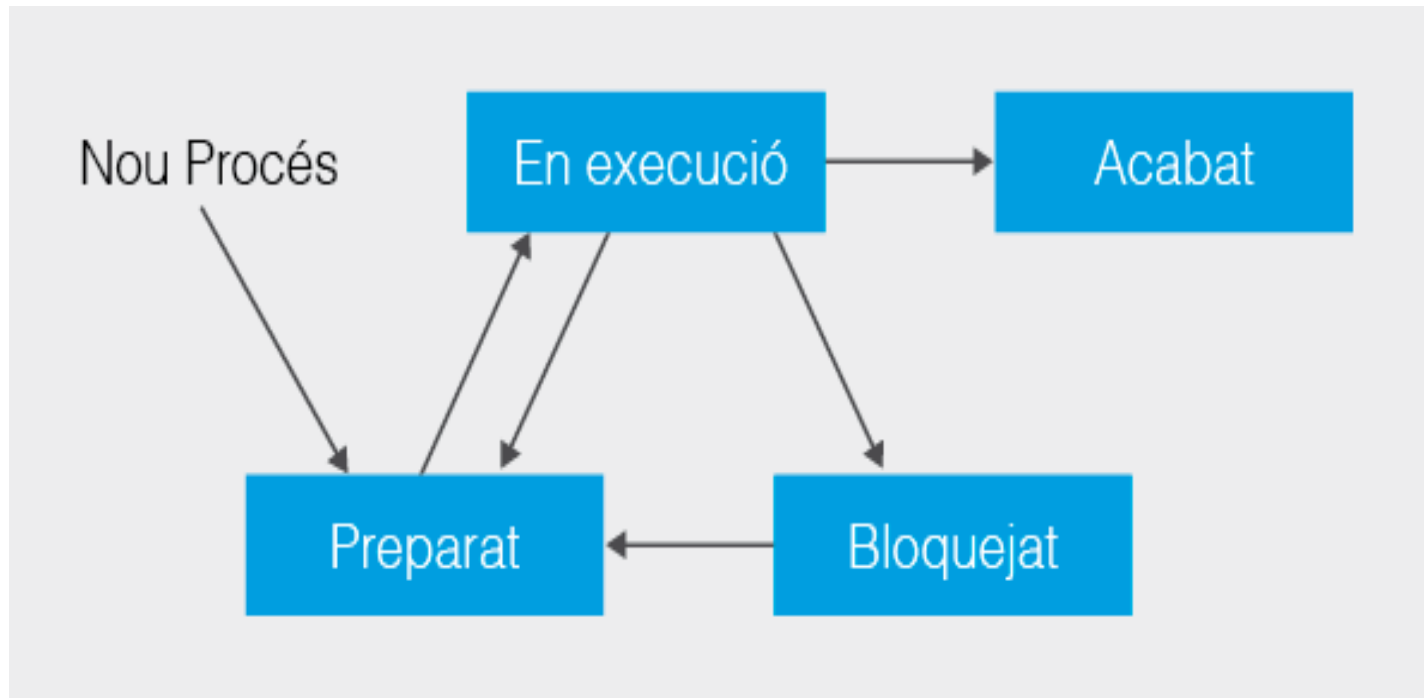
UF2: PROCESSOS I FILS NF1 Processos

Mòdul 9: PSP

Dionis Remón

3. Estats del proces

- El planificador de processos del sistema operatiu és el que decideix quan entra el procés a executar-se. Quan el procés s'està executat, el seu estat s'anomena en **execució**.
- Quan un procés abandoni el processador, perquè el planificador de processos així ho haurà decidit, canviarà a l'estat de preparat i es quedarà esperant que el planificador li assigni el torn per aconseguir de nou el processador i tornar a l'estat d'execució. Des de l'estat d'execució, un procés pot passar a l'estat de **bloquejat o en espera**.
- L'últim estat és acabat. És un estat al qual s'hi arriba un cop el procés ha finalitzat tota la seva execució i estarà a punt per tal que el sistema n'alliberi quan pugui els recursos associats.



4. Estats de transició

L'estat d'un procés es defineix per la seva activitat actual, i aquest va canviant a mesura que s'executa el procés.





1. Preparat-Execució: succeeix quan un procés es escollit pel planificador per a passar a execució.

2. Execució-Preparat: el procés en execució abandona involuntàriament el processador. Pot deures a que el procés ja ha consumit el temps que tenia assignat o a que un altre procés de major prioritat està preparat. En els dos casos el procés en execució s'interromp i és expulsat.

3. Execució-Suspès: el procés abandona voluntàriament el processador, ha d'esperar a que es produeixi un esdeveniment extern (finalització d'una operació d'E/S, per exemple).

4. Suspès-Preparat: l'esdeveniment al que estava esperant el procés suspès ha finalitzat, per tant pot passar novament a l'estat Preparat. Un procés mai pot passar directament de Suspès a Execució, ja que necessita aconseguir la CPU, i com que és un recurs més, ha d'esperar a que el planificador el trie.

5. Execució-Acabat: el procés ha acabat la seua execució (terminació normal) o bé ha estat eliminat (terminació anormal, per exemple causada per la recepció d'un senyal que no esperava).

Un procés podrà estar actiu (executant-se o esperant a la cua de preparats) o bloquejat (a l'espera d'un esdeveniment).

5. Bloc de Control de Procés

Conegut com BCP. La informació que es guarda és:

- Estat del procés
- Identificador del procés
- Direccions de memòria
- Informació dels fitxers que utilitza un procés
- Registres de la CPU: comptador del programa, registre de dades...
- Informació sobre planificació: algoritme emprat, prioritat del procés...
- Informació per a estadístiques: temps d'inici del procés, temps de CPU consumit...

6. Canvi de context

- Canvi de context: consisteix en treure-li la CPU al procés en execució i assignar-la a un altre procés que està en estat Preparat. Es produeix quan es processa una interrupció interna o externa.
- L'operació de canvi de context la realitza un component del SO denominat dispatcher.
- Els canvis de context poden suposar una sobrecàrrega en el sistema si es realitzen amb molta freqüència.



7. Avantatges de la multitasca

- Un SO multitasca permet estar executant diversos processos simultàniament.
- El SO s'encarrega de repartir el temps del processador entre aquests processos, per a que tots vagen avançant en la seua execució.
- Hui en dia, tret de situacions especials, tots els sistemes operatius suporten la multitasca.



- Avantatges:

- Facilita la programació ja que permet dividir les aplicacions en varis processos, beneficia la modularitat.
- Presta un bon servici, pot atendre a varis usuaris de forma eficient i simultània.
- Aprofita els temps morts que els processos passen esperant a que es completen les seues operacions d'E/S.
- Augmenta l'ús de la CPU ja que aprofita els espais de temps que els processos estan bloquejats.

8. Criteris de planificació

- Equitat: el processador assigna als processos de forma equitativa.
- Eficiència: que es mantingui la CPU ocupada el 100 % del temps.
- Temps de resposta: que es minimitza el temps de resposta.
- Temps d'espera en cua: minimitzar el temps en que els processos estan esperant en cua.
- Temps de procés global: minimitzar el temps total d'execució del procés.
- Rendiment: maximitzar el número de treballs processats per unitat de temps.

9. Algorismes de planificació

- FCFS (First Come First Served)
- SJF (Shortest Job First)
- Round Robin (Shortest Remaining Time First)
- Planificació per prioritat



FCFS (First Come First Served)

- És l'algorisme més senzill d'implementar.
- El planificador escull el procés que més temps porta a la cua de preparats.
- Dins dels seus desavantatges podem citar que processos amb ràfegues de CPU curtes poden estar esperant un temps elevat.

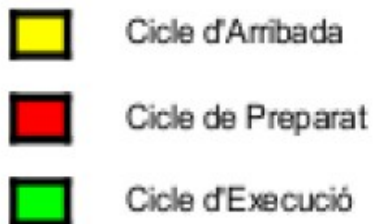
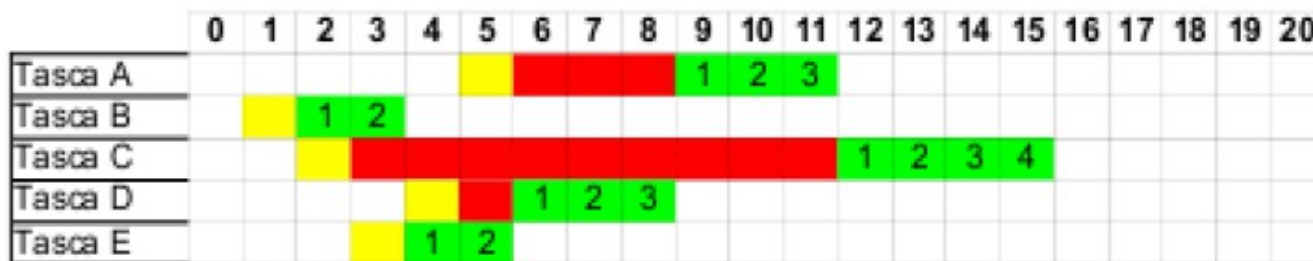
	Arribada	Cicles CPU	Inici	Fi
Tasca A	4	4	9	12
Tasca B	7	3	18	20
Tasca C	3	4	5	8
Tasca D	2	2	3	4
Tasca E	5	5	13	17



SJF (Shortest Job First)

- El planificador escull el procés que té un temps d'execució més curt de la ràfega de CPU següent.
- Si hi ha diversos processos amb temps previstos d'execució iguals a la cua de preparats, es pot aplicar una política FIFO per a seleccionar un d'ells.
- El principal problema que té es esbrinar quant dura la següent ràfega de CPU de cada procés, càlcul que s'acostuma a fer mitjançant estimacions basades en les anteriors execucions dels processos.

	Arribada	Cicles CPU	Inici	Fi
Tasca A	5	3	9	11
Tasca B	1	2	2	3
Tasca C	2	4	12	15
Tasca D	4	3	6	8
Tasca E	3	2	4	5



Round Robin (Shortest Remaining Time First)

- Sistemes de temps compartit.
- A cada procés se li assigna una unitat de temps denominada quantum durant el qual podrà executar-se.
- Els processos a la cua de preparats es gestionen amb una política FIFO.
- Quan el planificador escull el primer procés de la cua, aquest s'executa fins que esgota el seu quantum o acaba per que li quedava una ràfega de CPU menor que el quantum.

	Arribada	Cicles CPU	Inici	Fi
Tasca A	3	5	5	18
Tasca B	6	1	10	10
Tasca C	2	4	3	15
Tasca D	1	5	2	16
Tasca E	4	2	7	12



Planificació per prioritat

- La planificació per prioritat és més general que SJF, en que a cada tasca se li assigna una prioritat i la que té prioritat més alta passa per davant.
- A les prioritats són assignades amb enters.
- Prioritats poden ser assignades de manera externa o de manera interna.
 - Interna: Ve determinada per la SO
 - Externa: Per un usuari, per política d'empresa ...



Institut Poblenou

Consorci d'Educació
de Barcelona

Generalitat de Catalunya
Ajuntament de Barcelona

UF2: PROCESSOS I FILS

NF1 Processos

Mòdul 9: PSP

Dionis Remón

Exemple: Solaris

- És un sistema de prioritats amb quantums variables.
- La prioritat del procés és proporcional al nombre de quantums.
- Els processos que interactuen tenen prioritat més alta.
- Classes de planificació:
 - Prioritat fixa: per a certs processos la prioritat no es modifica dinàmicament
 - És planifica per conjunt de tasques anomenat projecte.
 - Hi ha una classe de prioritat d'ús exclusiu per a tasques del sistema.

priority	time quantum	time quantum expired	return from sleep
0	200	0	50
5	200	0	50
10	160	0	51
15	160	5	51
20	120	10	52
25	120	15	52
30	80	20	53
35	80	25	54
40	40	30	55
45	40	35	56
50	40	40	58
55	40	45	58
59	20	49	59

Figure 5.11 Solaris dispatch table for interactive and time-sharing threads.

Exemple: WinXP

- WinXP usa un algoritme de planificació preventiu basat en prioritats.
- El dispatcher usa un esquema de prioritats de 32 nivells. Hi ha tres classes:
 - Classe variable del 1 als 15
 - Classe de temps real del 16 al 31
 - Classe de gestió de memòria 0
- Hi ha un idle especials per a fils.

Exemple: WinXP

- WinXP identifica prioritat de 7 classes i 6 prioritats relatives.
- Els processos també se'ls assigna
- Els processos que estan en una finestra activa tenen el triple de quantums per a que vagi més fluid la interacció amb l'usuari.

	real-time	high	above normal	normal	below normal	idle priority
time-critical	31	15	15	15	15	15
highest	26	15	12	10	8	6
above normal	25	14	11	9	7	5
normal	24	13	10	8	6	4
below normal	23	12	9	7	5	3
lowest	22	11	8	6	4	2
idle	16	1	1	1	1	1

Figure 5.12 Windows XP priorities.

Bibliografia

- Abraham Silberschatz, Greg Gagne, and Peter Baer Galvin, "Operating System Concepts, Eighth Edition ", Capítol 5
- Apunts del IOC