

# Fonaments dels Sistemes Operatius (FSO)

Departament d'Informàtica de Sistemes i Computadores (DISCA)  
*Universitat Politècnica de València*

## Bloc Temàtic 4: Gestió de Memòria Unitat Temàtica 9

### Gestió de Memòria

f SO

DISCA

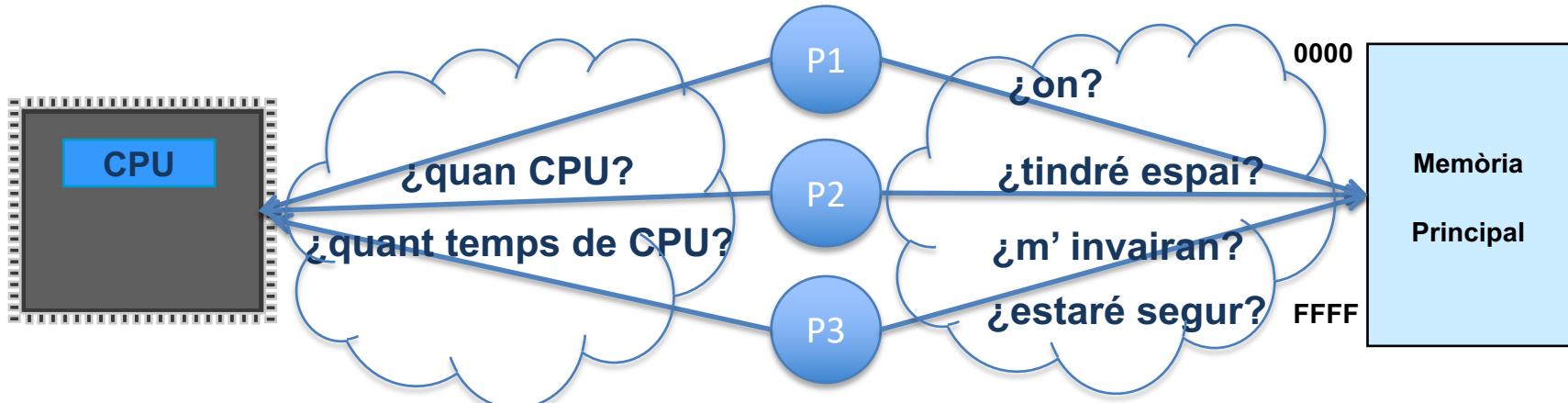


UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

- **Objectius**
  - Introduir els conceptes bàsics **de gestió de memòria**
  - Comprendre la diferència entre **memòria lògica i física**
  - Entendre el concepte d'**assignació contígua de memòria**
  - Analitzar la problemàtica de **fragmentació** associada a la assignació contigua
  - Estudiar **estratègies d'assignació contigua**
- **Bibliografia:**
  - “Fundamentos de sistemas operativos” (capítulo-8)  
Silberschatz 7<sup>a</sup> Ed
  - “Sistemas operativos: una visión aplicada” (capítulo-5) Jesús Carretero 2<sup>º</sup> Ed

- **Contingut**
  - Introducció
  - Problemàtica de gestió de memòria
  - Adreces lògiques vs adreces físiques
  - Unitat de maneig de memòria (MMU)
  - Assignació contigua de memòria
    - Particions fixes
    - Particions variables
      - First Fit
      - Best Fit
      - Worst Fit
      - Fragmentació externa i compactació

- Per a **executar programes**
  - Tant les **instruccions** com les **dades** han de trobar-se en **Memòria Principal**
- Per a augmentar l'**eficiència del processador** i la seu velocitat de resposta → Sistema multiprogramat
  - Els processos d'**un sistema multiprogramat**
    - Comparteixen CPU's → **Planificador de processos**
    - Comparteixen Memòria Principal → **Gestor de la Memòria**



- **Memòria Principal**

- Consisteix en una gran **matriu de paraules o bytes** cadascuna d'elles **amb** la seuà pròpia **adreça física** associada
- És un **recurs crític**
  - **Temps d'accés a memòria:** és crucial per al funcionament del sistema, ja que s'accedeix continuament. En el cicle d'execució d'una instrucció, es llig la instrucció, les dades i s'emmagatzemem resultats
  - **Grandària de memòria:** La memòria té una capacitat d'emmagatzemament limitat

- Capacitat d'emmagatzemar informació **volàtil** del computador

- Existeixen diversos nivells d'emmagatzemament o memòria:
  - **Registres** de CPU
  - Memòria **cache**: buffer que permet resoldre la diferència de velocitat entre accessos a memòria i freqüència de CPU
  - Memòria principal (**RAM**)

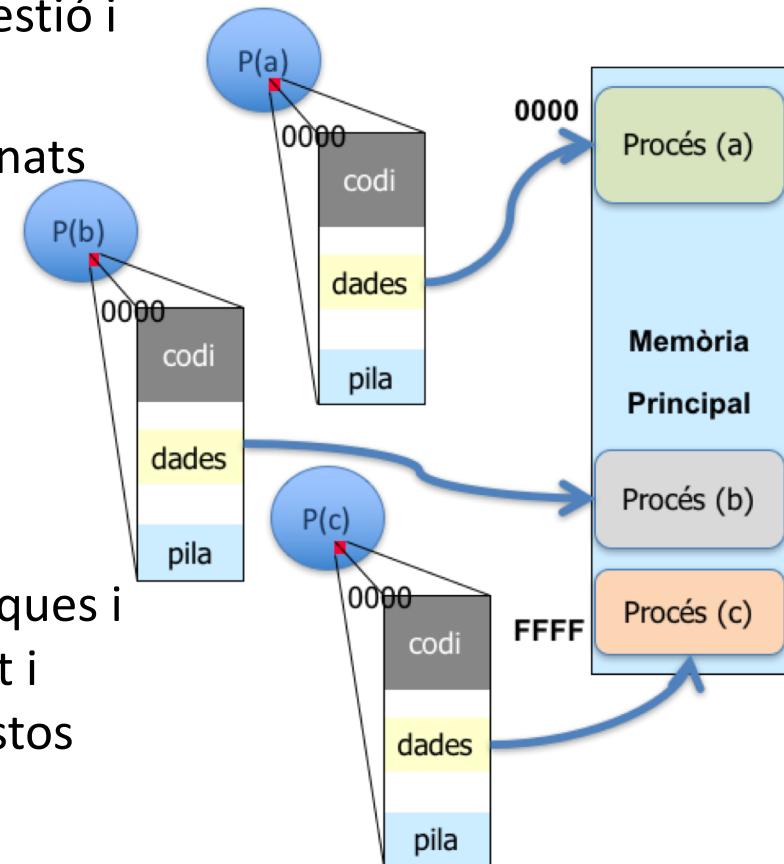


- Introducció
- **Problemàtica de gestió de memòria**
- Adreces lògiques vs adreces físiques
- Unitat de maneig de memòria (MMU)
- Assignació contigua de memòria

# Problemàtica de gestió de memòria

fso

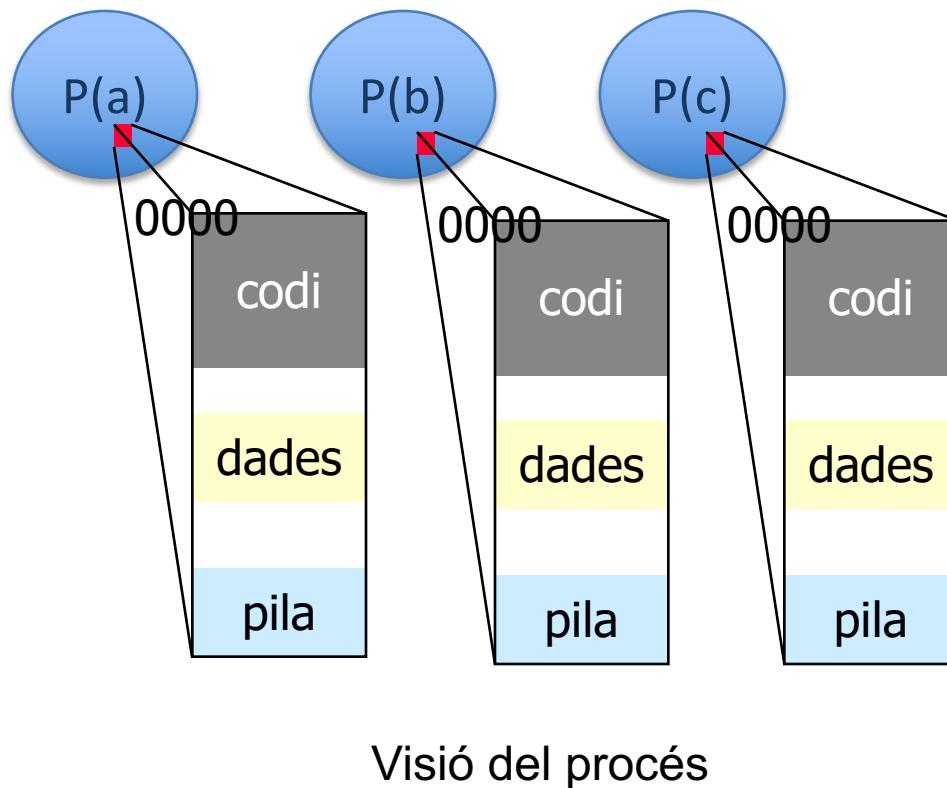
- El Sistema Operatiu ha d'ofrir una bona gestió i ús eficient de la memòria.
- Per a fer-ho cal resoldre problemes relacionats amb
  - **Protecció**
  - **Mancança**
  - **Assignació**
  - **Reubicació**
- Els sistemes actuales estan dotats de tècniques i mecanismes diversos que han evolucionat i millorat amb el temps per a resoldre aquests problemes
  - Espai d'adreses lògic → Reubicació
  - MMU → Reubicació i protecció
  - Biblioteques dinàmiques → Mancança
  - Memòria virtual → Mancança
  - Técniques d'assignació → Assignació



- Introducció
- Problemàtica de gestió de memòria
- **Adreces lògiques vs adreces físiques**
- Unitat de maneig de memòria (MMU)
- Assignació contigua de memòria

- **Adreces lògiques**

- Espai d'adreçament independent per a cada procés
- Permeten que el codi no s'involucra en particularitats de la màquina-> reubicació

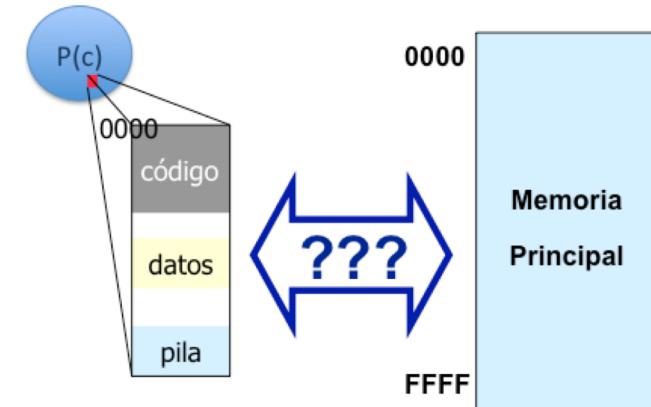


- Necessitat de traduir adreces lògiques a físiques

¿Quina adreça física correspon a una adreça lògica?

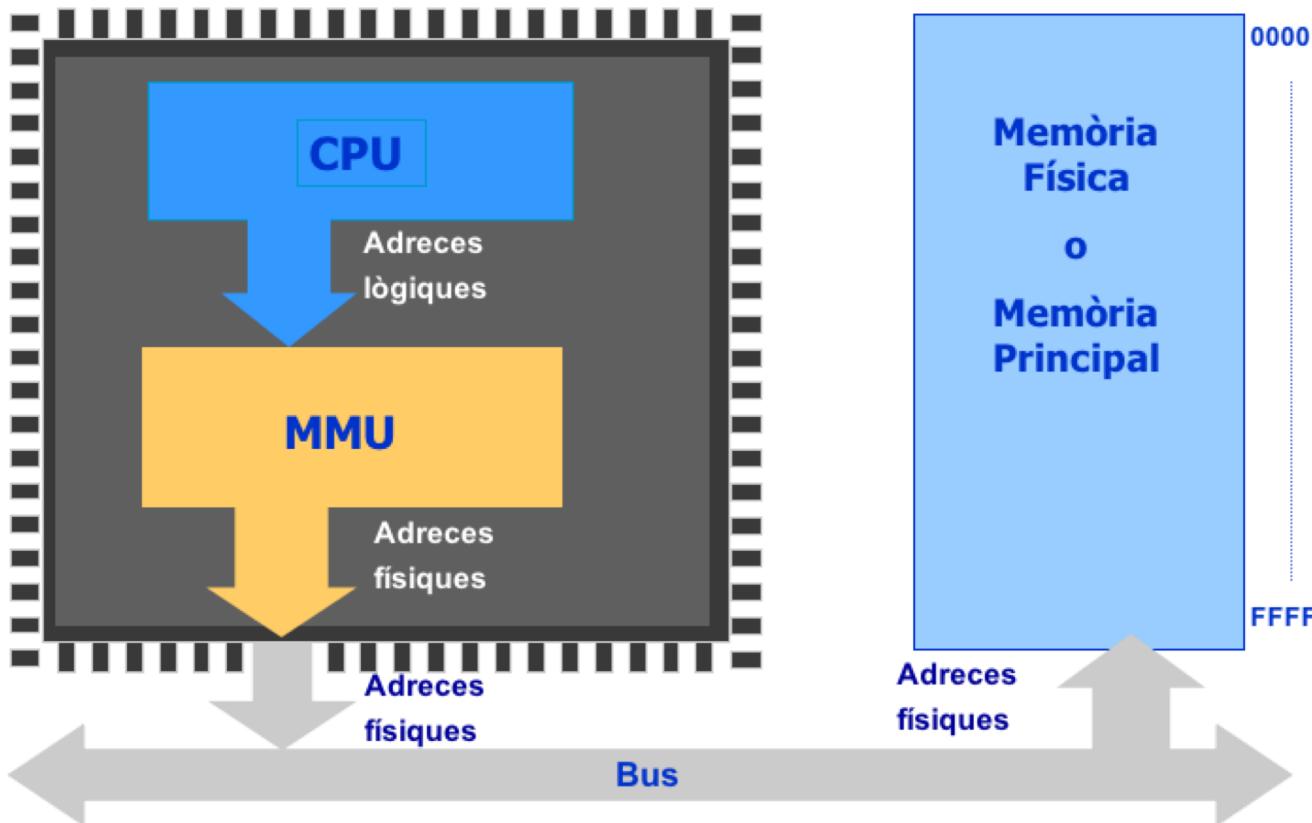
- Utilitzant un espai d'adreces lògiques (AL) i altre físic (AF)  
és necessari:

- Establir una funció de traducció d'AL a AF
- Decidir **com** s'implementa aquesta funció
  - Hardware
  - Software
- **Quan** es duu a terme la traducció?
  - **En compilar el programa**
    - » Codi absolut . No reubicable
  - **En carregar** el programa en memòria
    - » Codi reubicable en temps de càrrega
  - **En executar** cada instrucció
    - » **Codi reubicable en temps execució**
    - » El procés pot desplaçar-se d'una posició a una altra de memòria mentre s'executa

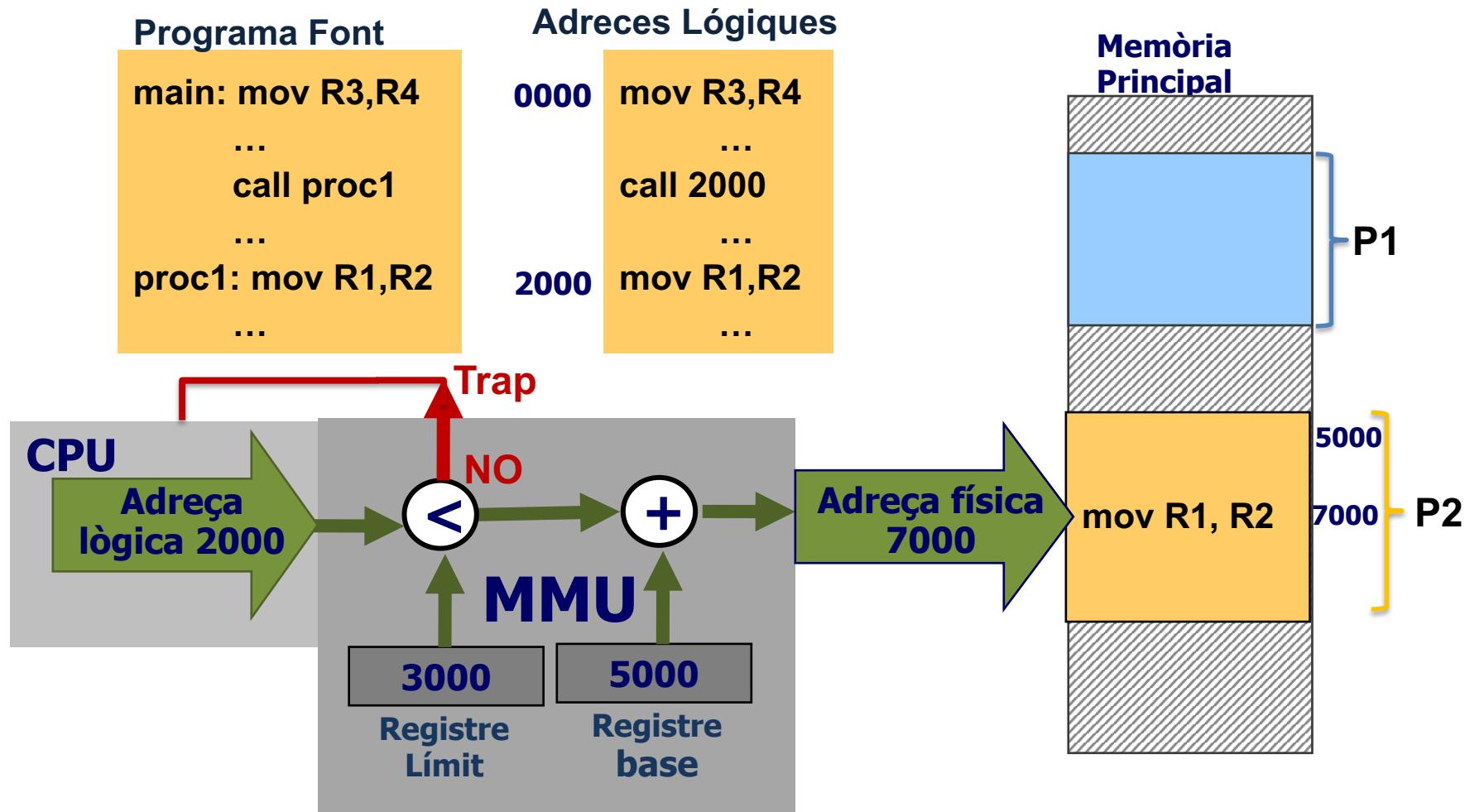


- Introducció
- Problemàtica de gestió de memòria
- Adreces lògiques vs adreces físiques
- **Unitat de maneig de memòria (MMU)**
- Assignació contigua de memòria

- **Traduir adreces lògiques a adreces físiques ralentitza l'execució d'instruccions → sobrecàrrega**
  - S'aconsegueix major velocitat de traducció amb hardware específic
    - MMU → unitat de maneig de memòria



- Model senzill de MMU
  - basat en **registre base** i **registre límit**



!!!!!! Garantides la reubicació en temps d'execució i la protecció

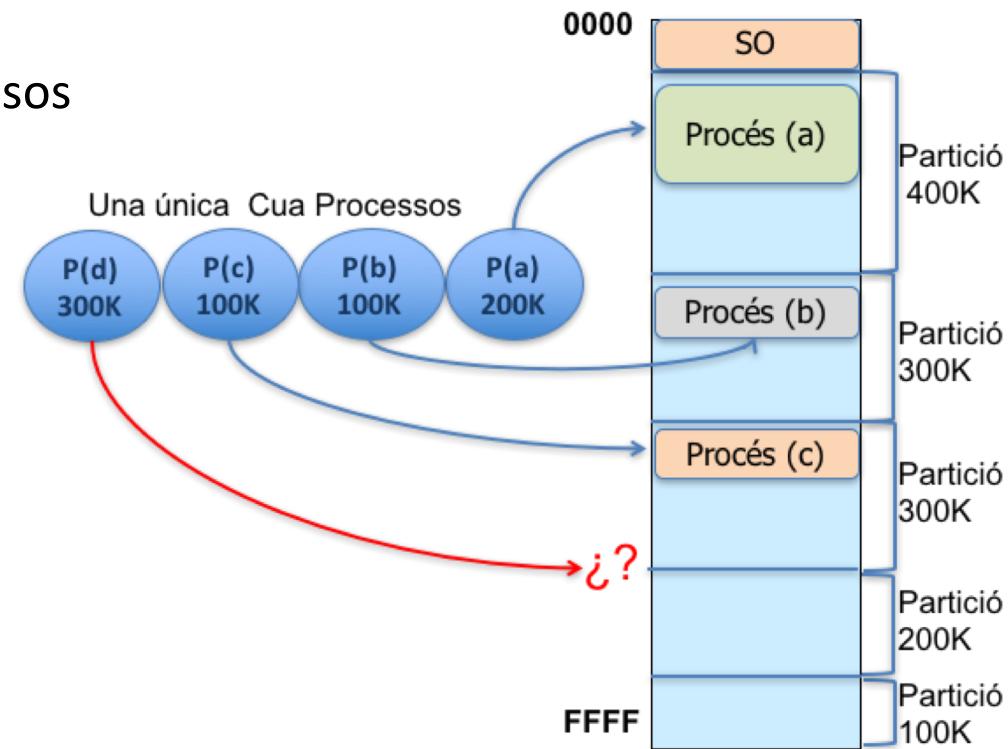
- Introducció
- Problemàtica de gestió de memòria
- Adreces lògiques vs adreces físiques
- Unitat de maneig de memòria
- **Assignació contigua de memòria**

- Model de **gestió de memòria contigua**
  - Un **procés** està ubicat en memòria en una única secció d'**adreces físiques** contigües
- La memòria se sol trobar dividida en dues parts:
  - La zona on resideix el Sistema Operatiu
  - La zona on resideixen els processos d'usuari
- Estratègies d'assignació contigua:
  - **Particions fixes:** definir a priori (per configuració del sistema) les seccions de memòria on ubicar els processos
  - **Particions variables:** que el sistema pot ubica els processos en els buits de memòria que es troben lliures

# Assignació contigua de memòria

## • Particions fixes

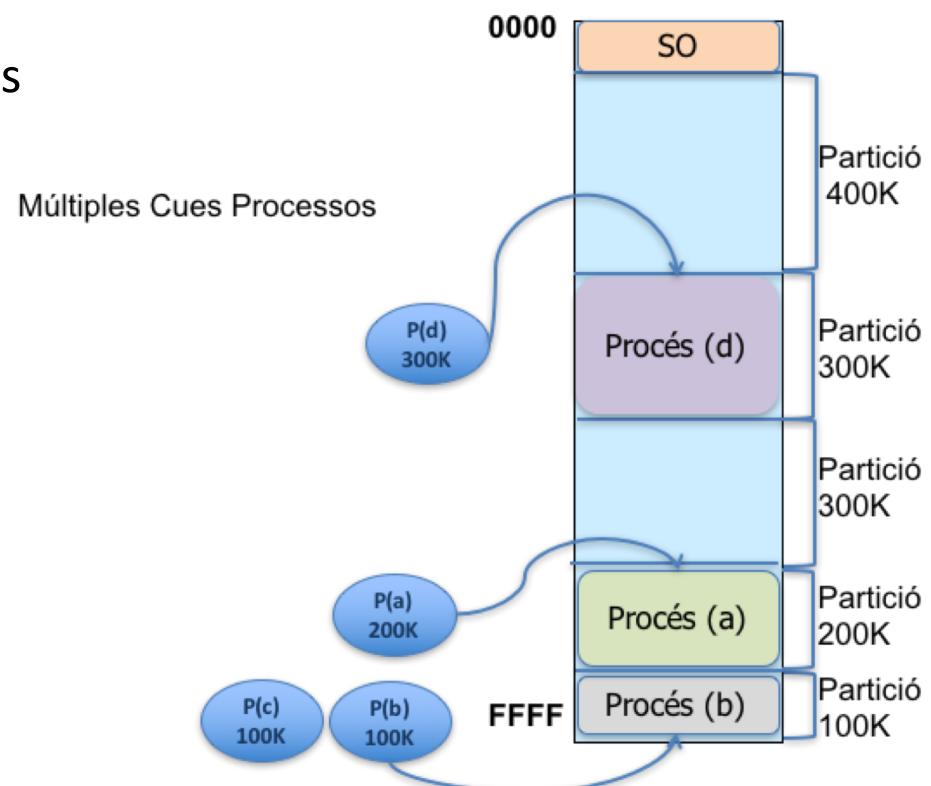
- La memòria se suposa dividida en particions de diferents grandàries fixes
- El sistema operatiu manté una **llista de particions lliures**
- Variants
  - Cua única de processos
  - Múltiples cues de processos
- Problema:
  - **Fragmentació interna**



# Assignació contigua de memòria

## • Particions fixes

- La memòria se suposa dividida en particions de diferents grandàries fixes
- El sistema operatiu manté una **llista de particions lliures**
- Variants
  - Cua única de processos
  - Múltiples cues de processos
- Problema:
  - **Fragmentació interna**



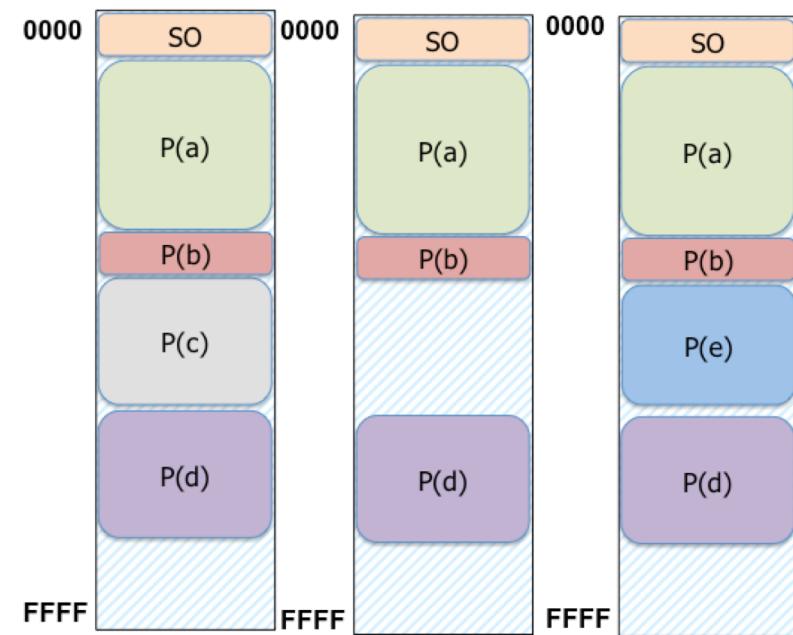
# Assignació contigua de memòria

- Particions variables

- Inicialment la memòria disponible per a processos està tota lliure en un **únic buit**
- A mesura que apleguen peticions de processos es va assignant memòria
- El sistema operatiu manté una **llista de buits que conté la seua adreça inicial i la seua grandària.**



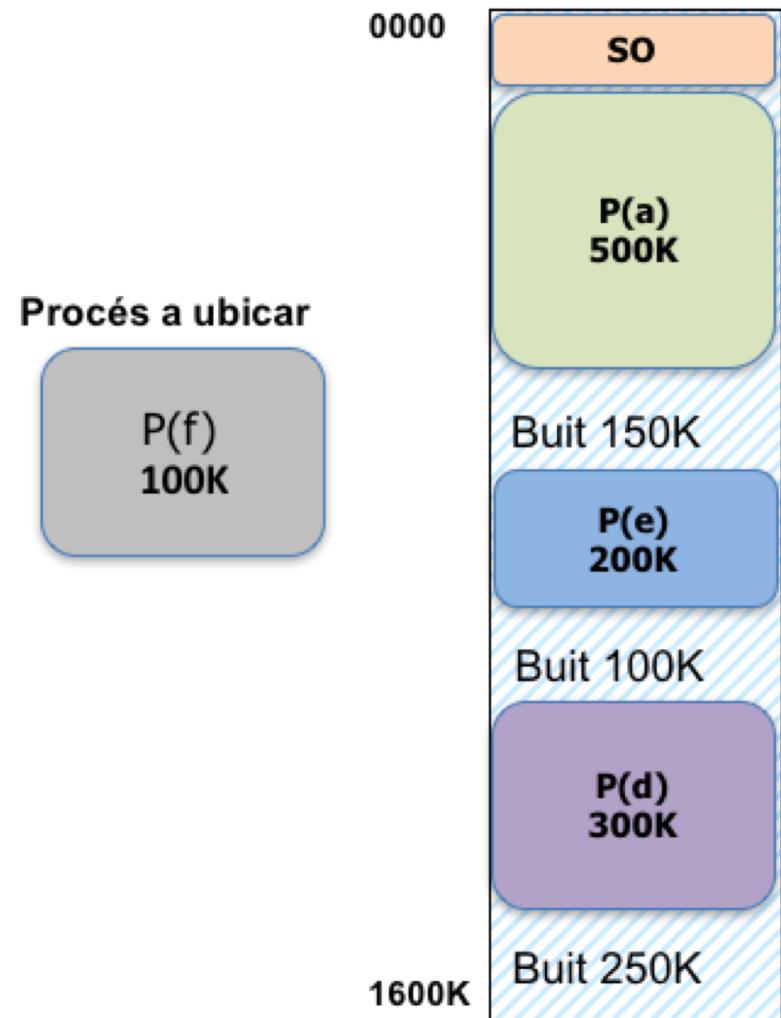
- Problema:
  - Fragmentació Externa**
- Estratègies d'assignació de buits:
  - Millor buit (Best Fit)**
    - Se ubica el procés en el menor buit que satisfà la seu petició
  - Pitjor buit (Worst Fit)**
    - Se ubica el procés en el major buit que satisfà la seu petició
  - Primer buit (First Fit)**
    - Se ubica el procés en el primer buit que satisfà la seu petició



# Assignación Contigua

- Particions Variables

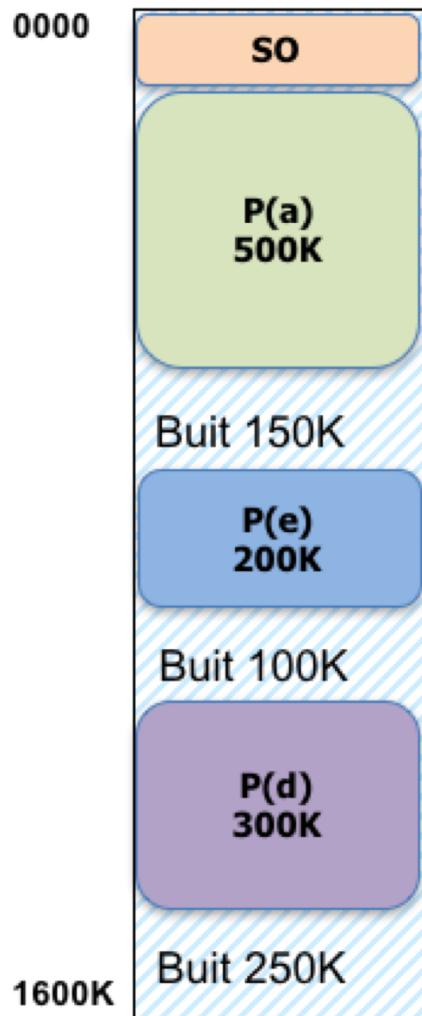
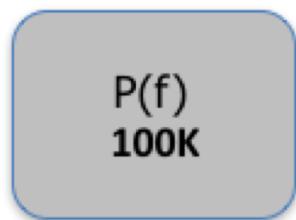
¿On s'ubicaria el procés  $P(f)$  en cadascuna de les estratègies d'assignació de buits?



# Assignació Contigua

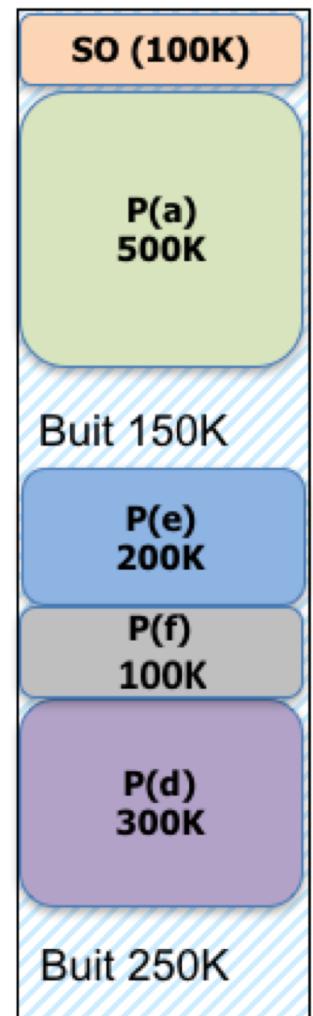
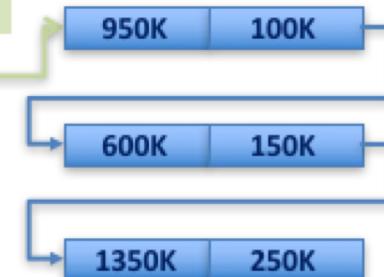
- Estratègia Millor Ajust

Procés a ubicar



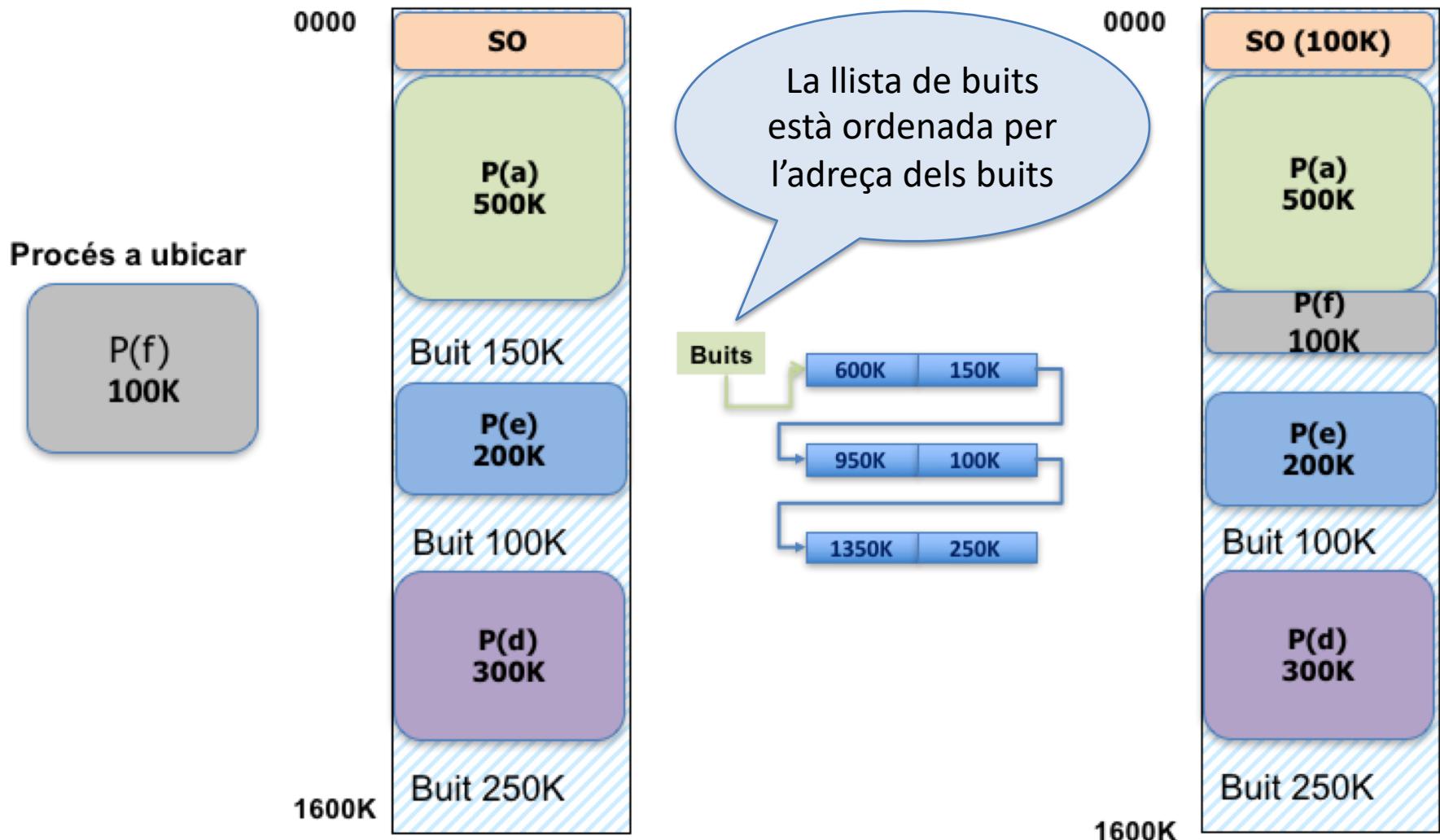
La llista de buits  
està ordenada per  
grandària de buit

Buits



# Assignació Contigua

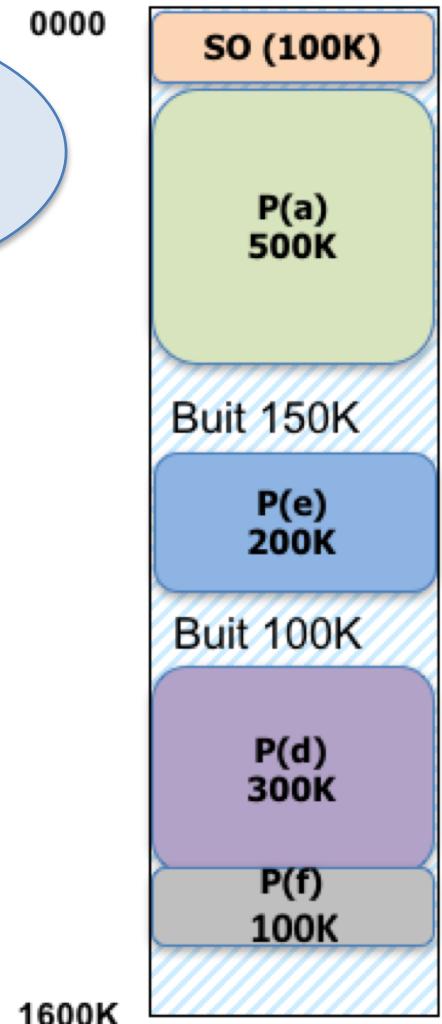
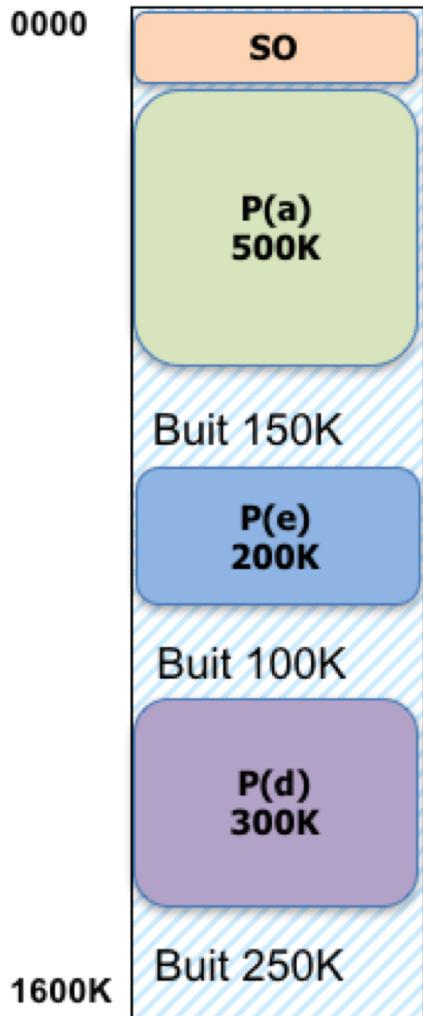
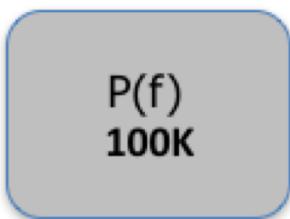
- Estratègia Primer Ajust



# Assignació Contigua

- Estratègia Pitjor Ajuste

Procés a ubicar

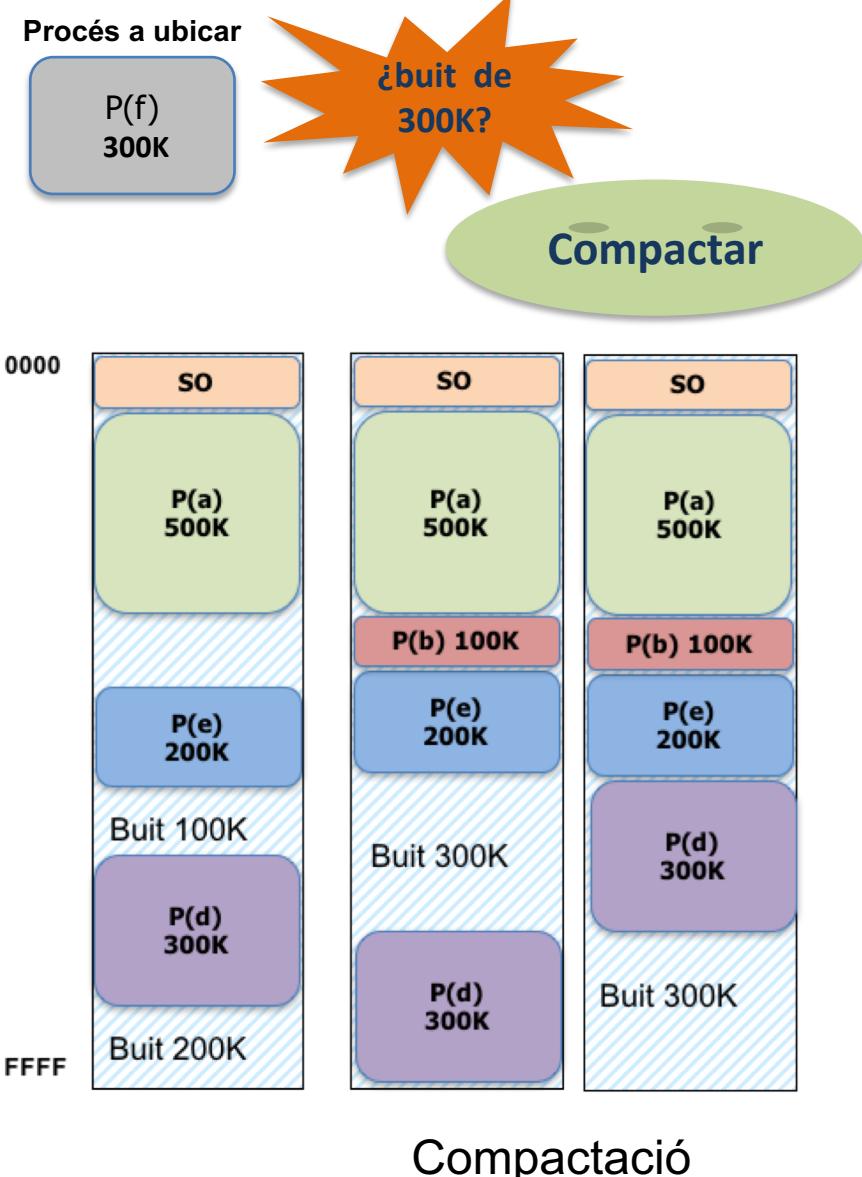


- **Fragmentació externa**

- Sumant les grandaries dels buits existents hi hauria prou memòria lliure per a satisfer una demanda, però al no estar contigus no se li pot assignar al procés

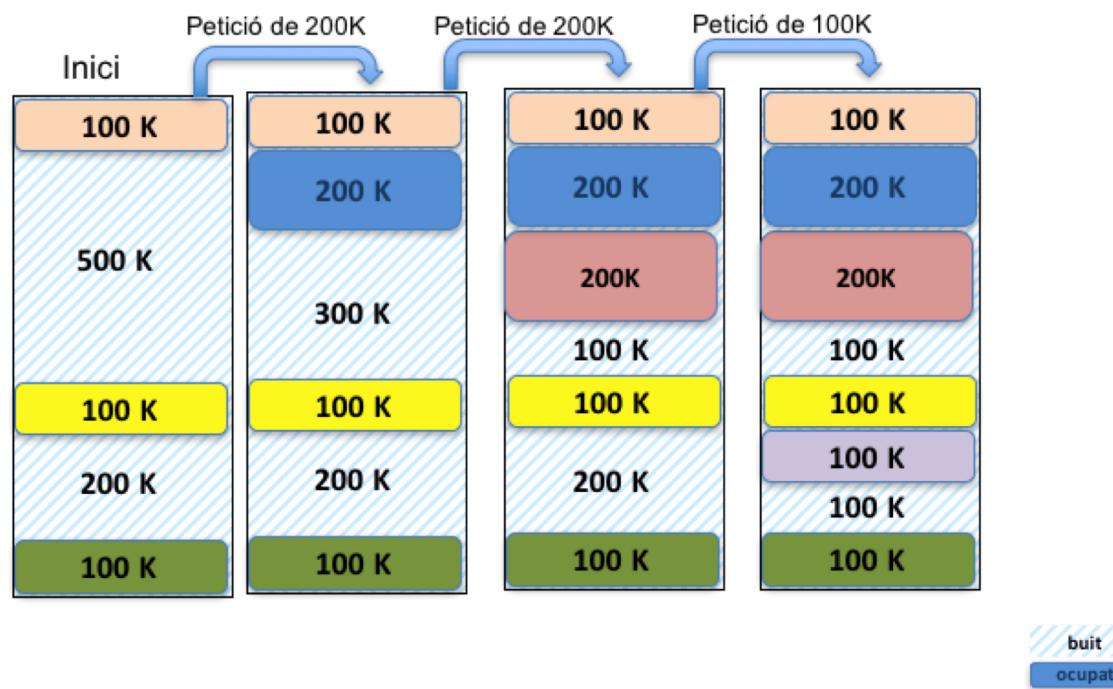
- **Compactació**

- Solució per a la fragmentació externa
- Els processos es reubiquen en memòria
- Imprescindible codi reubicable en temps de execució per a compactar



# Exercici 7.1:

- Siga un sistema operatiu que gestiona la memòria mitjançant **assignació contigua amb particions dinàmiques (variables)**. Partint d'un estat de la memòria indicat com Inici, l'ocupació de la mateixa front les peticions realitzades ha sigut la que es representa en els gràfics:



- Determine i justifique l'algorisme utilitzat entre millor ajust, pitjor ajust i primer ajust per a l'assignació de les demandes.

- Un sistema operatiu gestiona la seu memòria de 8196 paraules mitjançant **assignació contigua amb particions variables**. En un instant t el sistema està executant tres processos de grandària:

Procés A de grandària 1024 paraules

Procés B de grandària 3072 paraules

Procés C de grandària 3584 paraules

Racone si els següents escenaris independents entre sí són possibles:

- Que el procés B genere l'adreça lògica 960 i que aquesta siga traduïda per l'adreça física 725
- Que el procés A genere l'adreça lògica 1500 i que ésta sea traduïda per l'adreça física 1500
- Que el procés C en l'instant t genere l'adreça lògica 525, que se traduisca per l'adreça física 2061 i en el instant t+10 la mateixa adreça lògica es traduisca per l'adreça física 1549

- Memòria d'assignació contigua
  - Objectes interactius per a aprenetatge
    - [http://labvirtual.webs.upv.es/Fijas Multiples colas.htm](http://labvirtual.webs.upv.es/Fijas_Multiples_colas.htm)
    - [http://labvirtual.webs.upv.es/Fijas Una colas.htm](http://labvirtual.webs.upv.es/Fijas_Una_colas.htm)
    - [http://labvirtual.webs.upv.es/Best Fit.htm](http://labvirtual.webs.upv.es/Best_Fit.htm)
    - [http://labvirtual.webs.upv.es/Worst Fit.htm](http://labvirtual.webs.upv.es/Worst_Fit.htm)