

Fundamentos de los Sistemas Operativos (FSO)

Departamento de Informática de Sistemas y Computadoras (DISCA)
Universitat Politècnica de València

Bloque Temático 4: Gestión de Memoria
Unidad Temática 9

Gestión de Memoria

f SO

DISCA

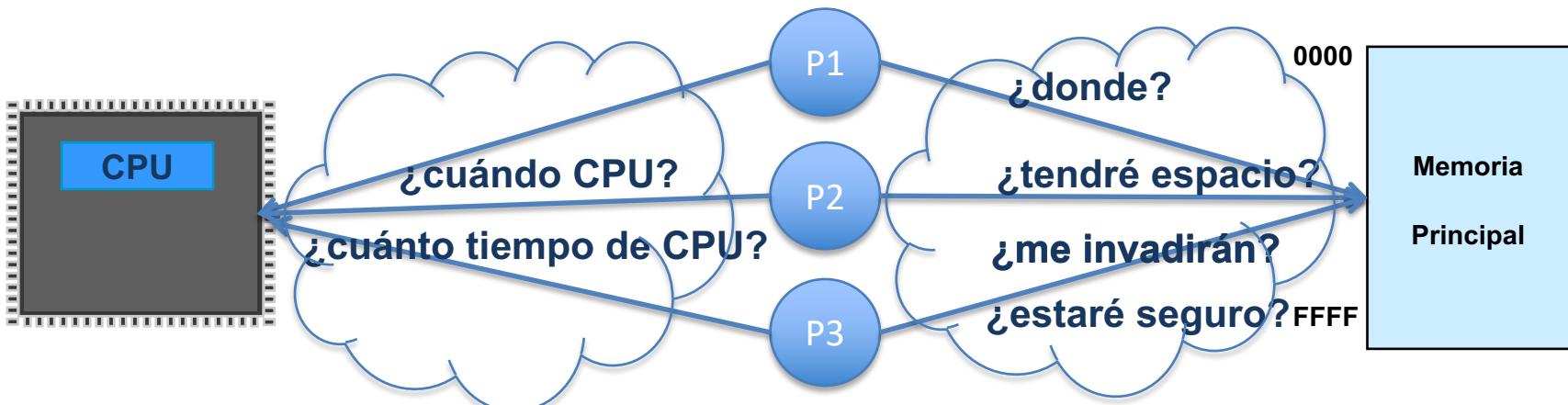


UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

- **Objetivos**
 - Introducir los conceptos básicos **de gestión de memoria**
 - Comprender la diferencia entre **memoria lógica y física**
 - Entender el concepto de **asignación contigua de memoria**
 - Analizar la problemática de **fragmentación** asociada a la asignación contigua
 - Estudiar **estrategias de asignación contigua**
- **Bibliografía:**
 - “Fundamentos de sistemas operativos” (capítulo-8)
Silberschatz 7^a Ed
 - “Sistemas operativos: una visión aplicada” (capítulo-5) Jesús Carretero 2^º Ed

- **Introducción**
- Problemática de gestión de memoria
- Direcciones lógicas vs direcciones físicas
- Unidad de manejo de memoria (MMU)
- Asignación contigua de memoria
 - Particiones fijas
 - Particiones variables
 - First Fit
 - Best Fit
 - Worst Fit
 - Fragmentación externa y compactación

- Para ejecutar programas: sus **instrucciones y datos** deben ubicarse en **Memoria Principal**
- Sistemas multiprogramados: aumentan la **utilización del procesador** y **tasa de rendimiento**
- Los procesos de **un sistema multiprogramado**
 - Comparten CPU's → **Planificador de procesos**
 - Comparten Memoria Principal → **Gestionar la Memoria**



- **Memoria Principal**

- Consiste en una gran **matriz de palabras o bytes** cada uno con su propia **dirección física** asociada
- Es un **recurso crítico**
 - **Tiempo de acceso a memoria:** es crucial para el funcionamiento del sistema, ya que se accede continuamente. En el ciclo de ejecución de una instrucción, se lee la instrucción, los datos y se almacenan resultados
 - **Tamaño de memoria:** La memoria tiene capacidad de almacenamiento **limitada**

- Capacidad de almacenar información **volátil** en el computador

- Existen varios niveles de almacenamiento o memoria:
 - **Registros** de CPU
 - **Memoria cache:** buffer que permite resolver la diferencia de velocidad entre accesos a memoria y frecuencia de CPU
 - **Memoria principal (RAM)**

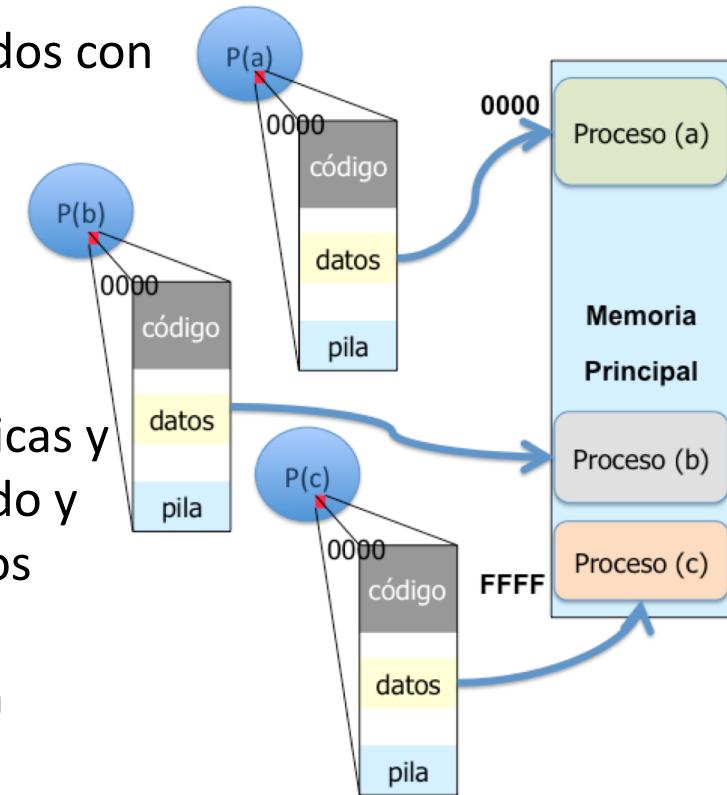


- Introducción
- **Problemática de gestión de memoria**
- Direcciones lógicas vs direcciones físicas
- Unidad de manejo de memoria (MMU)
- Asignación contigua de memoria

Problemática de gestión de memoria

fso

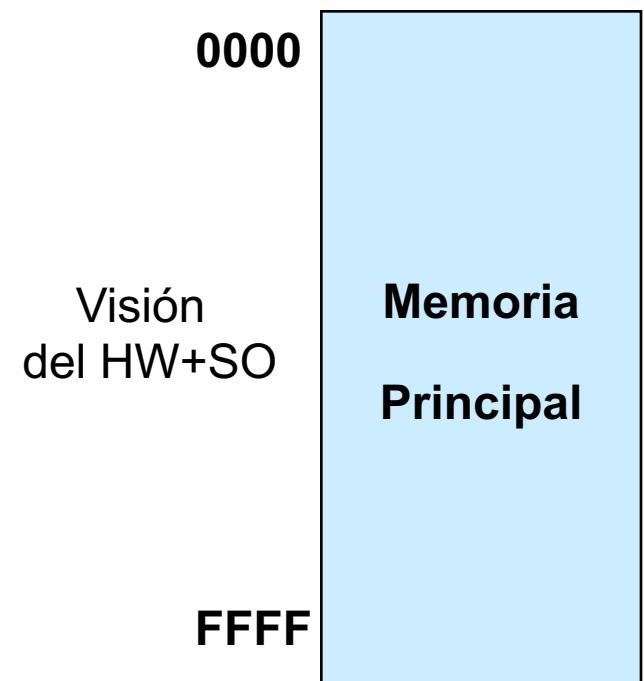
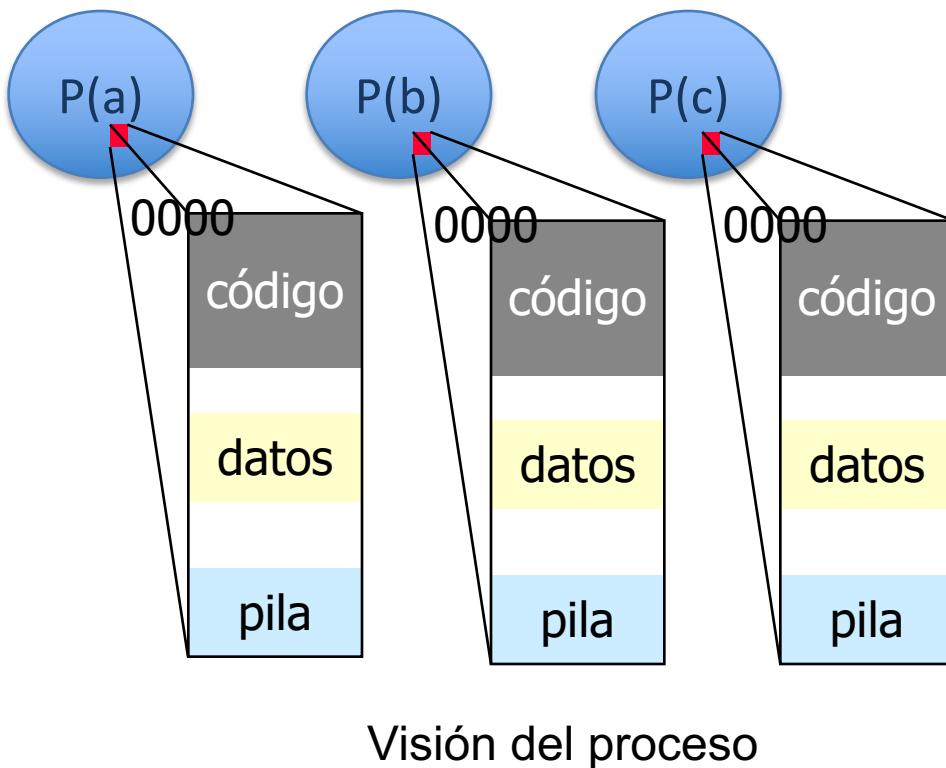
- El Sistema Operativo debe ofrecer una buena gestión y uso eficiente de la memoria
- Para ello debe resolver problemas relacionados con
 - **Protección**
 - **Escasez**
 - **Asignación**
 - **Reubicación**
- Los sistemas actuales están dotados de técnicas y mecanismos diversos que han evolucionando y mejorando con el tiempo para resolver dichos problemas
 - Espacio de direcciones lógico → Reubicación
 - MMU → Reubicación y protección
 - Librerías dinámicas → Escasez
 - Memoria Virtual → Escasez
 - Técnicas de Asignación → Asignación



- Introducción
- Problemática de gestión de memoria
- **Direcciones lógicas vs direcciones físicas**
- Unidad de manejo de memoria (MMU)
- Asignación contigua de memoria

- **Direcciones lógicas**

- Espacio de direccionamiento independiente para cada proceso
- Permiten que el código no se involucre en particularidades de la máquina -> reubicación

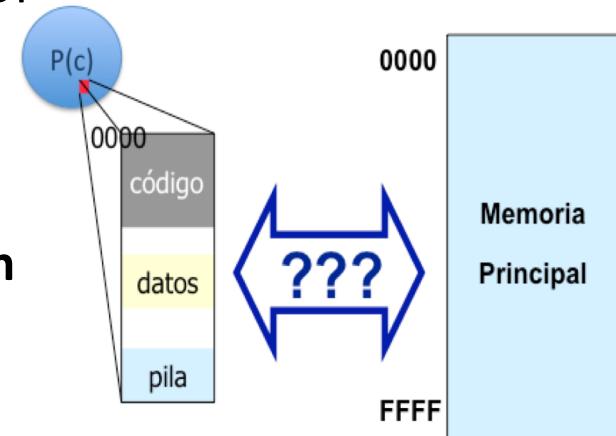


- Necesidad de traducir direcciones lógicas a físicas

¿Qué dirección física corresponde a una dirección lógica?

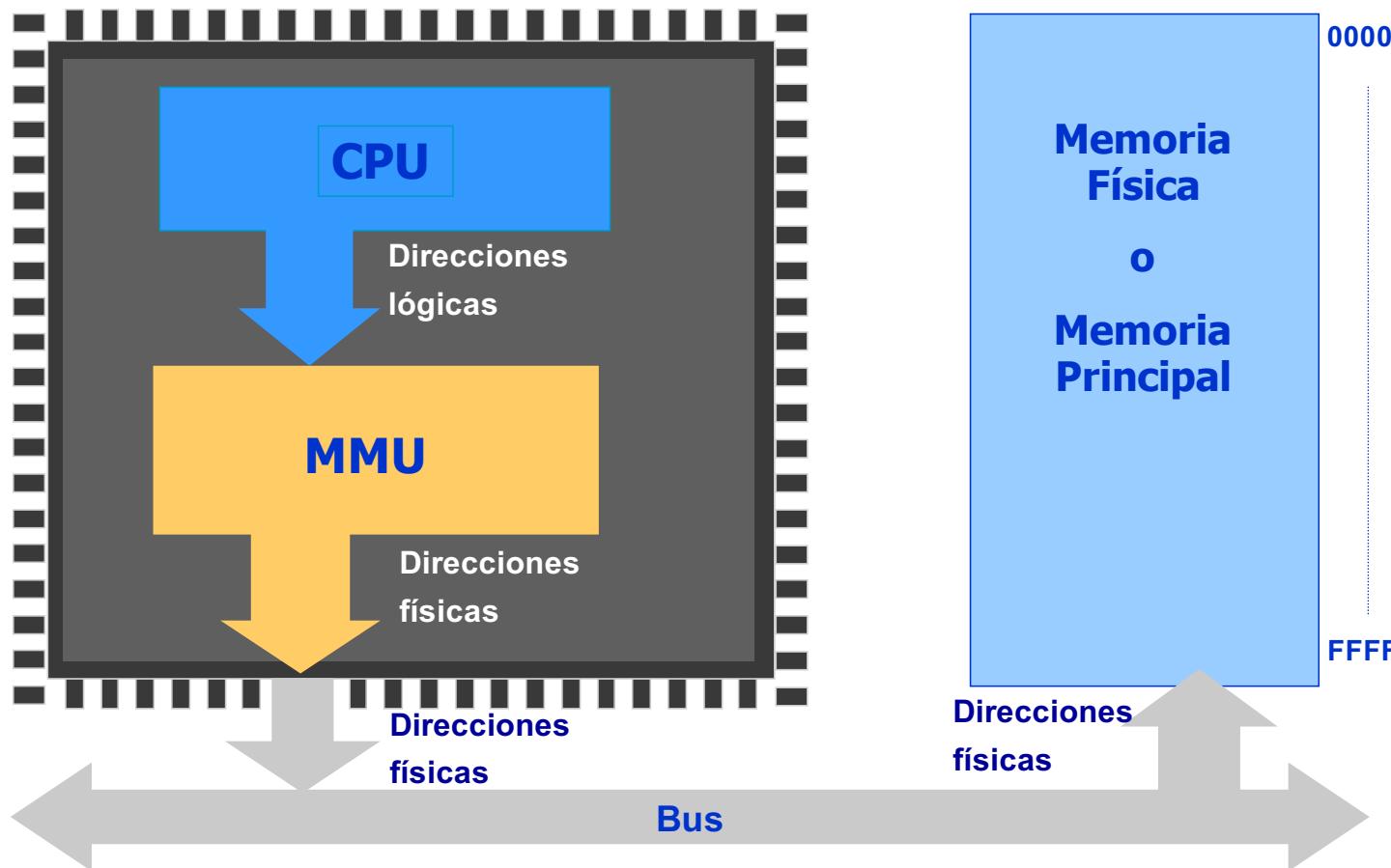
- Utilizando un espacio de direcciones lógico (DL) y otro físico (DF) es necesario:

- Establecer la función de traducción de DL a DF
- Decidir **cómo** se implementa dicha función
 - Hardware
 - Software
- Decidir **cuándo** se lleva a cabo la traducción
 - Al **compilar** el programa
 - » Código absoluto . **No reubicable**
 - Al **cargar** el programa en memoria
 - » Código reubicable en tiempo de carga
 - Al **ejecutar** cada instrucción
 - » **Código reubicable en tiempo ejecución**
 - » El proceso puede desplazarse de una posición a otra de memoria mientras se ejecuta



- Introducción
- Problemática de gestión de memoria
- Direcciones lógicas vs direcciones físicas
- **Unidad de manejo de memoria (MMU)**
- Asignación contigua de memoria

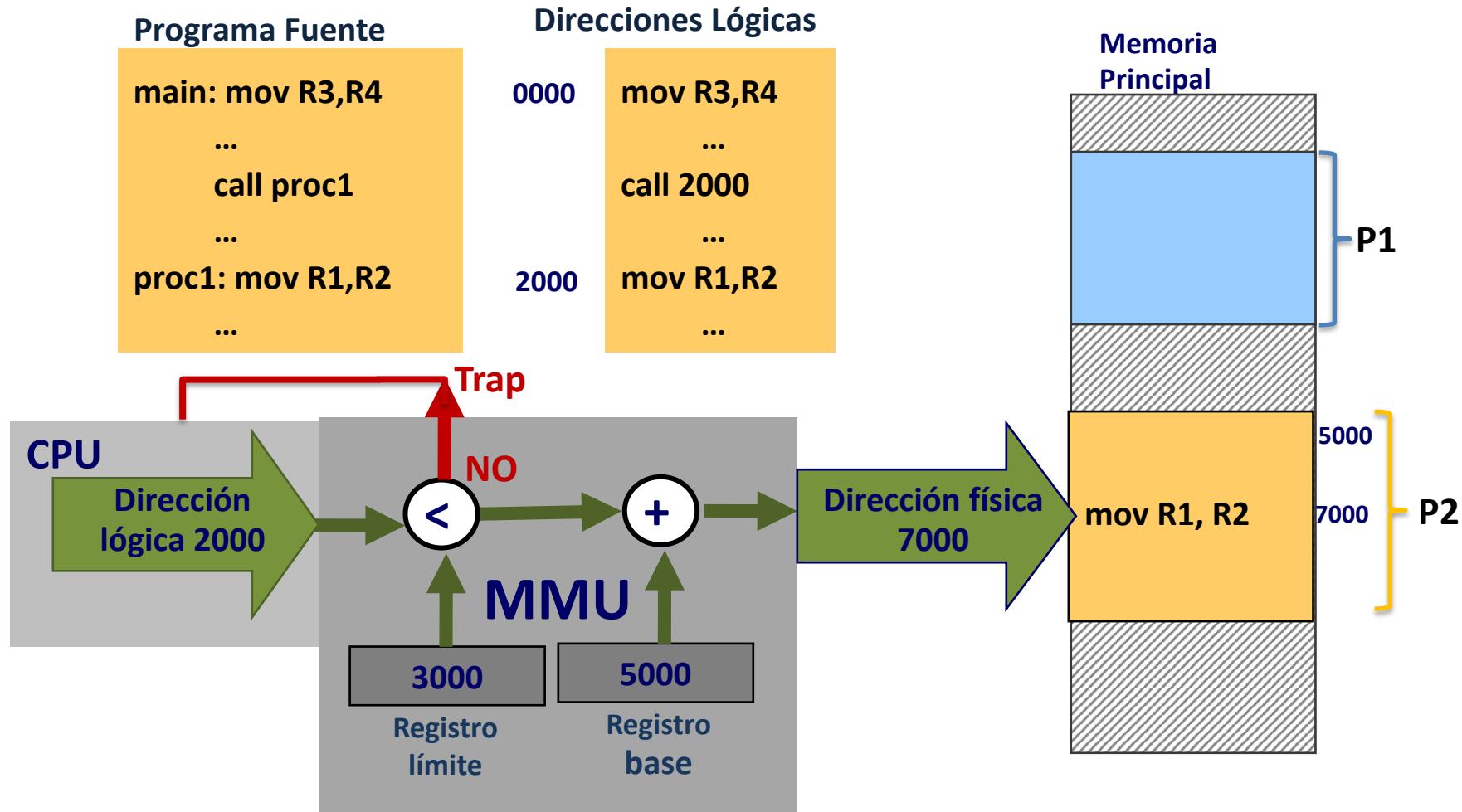
- Traducir direcciones lógicas a direcciones física ralentiza la ejecución de instrucciones y sobrecarga el sistema
 - Se consigue mayor velocidad de traducción con hardware específico
 - MMU → unidad de manejo de memoria



Unidad de manejo de memoria (MMU)

fso

- Modelo sencillo de MMU
 - Basado en **registro base** y **registro límite**



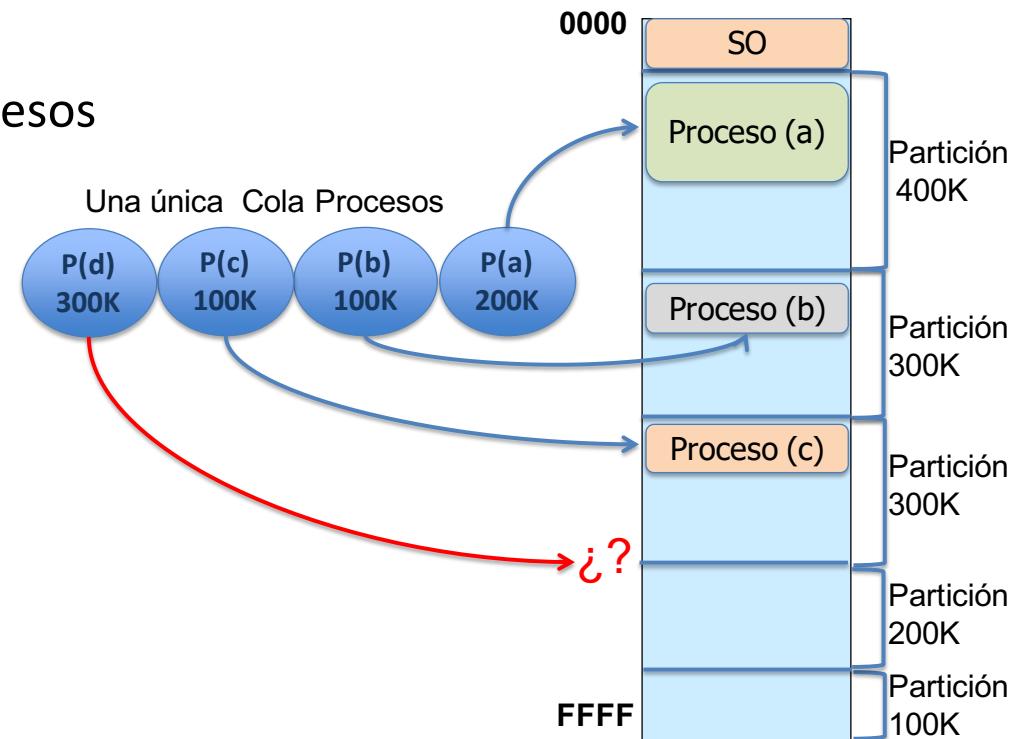
!!!!!! Garantizadas la reubicación en tiempo de ejecución y la protección

- Introducción
- Problemática de gestión de memoria
- Direcciones lógicas vs direcciones físicas
- Unidad de manejo de memoria
- **Asignación contigua de memoria**

- **Modelo de gestión de memoria contigua**
 - Un **proceso** está ubicado en memoria en una única sección de **direcciones físicas** contiguas
- Memoria principal suele estar dividida en dos partes:
 - La zona donde reside el Sistema Operativo
 - La zona donde residen los procesos de usuario
- Estrategias de asignación contigua:
 - **Particiones fijas:** Se define a priori (por configuración del sistema) el tamaño de las secciones de memoria donde ubicar los procesos
 - **Particiones variables:** el sistema puede ubicar los procesos en los huecos de memoria que estén libres

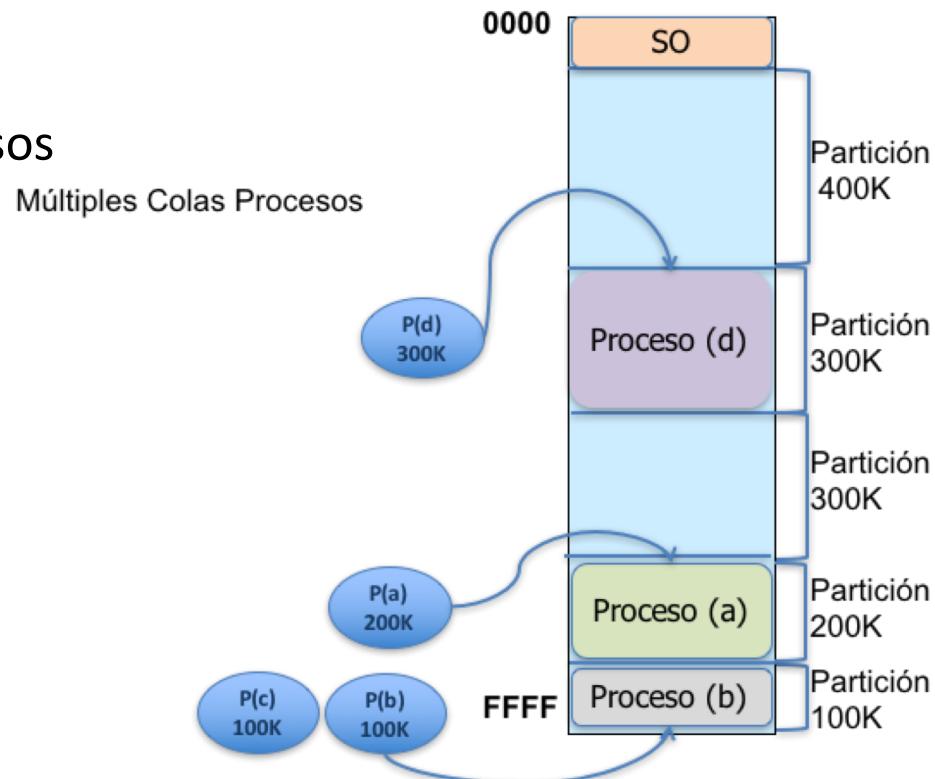
- **Particiones fijas**

- La memoria se supone dividida en particiones de diferentes tamaños fijos
- El sistema operativo mantiene una **lista de particiones libres**
- Variantes
 - Cola única de procesos
 - Múltiples colas de procesos
- Problema:
 - **Fragmentación interna**



- **Particiones fijas**

- La memoria se supone dividida en particiones de diferentes tamaños fijos
- El sistema operativo mantiene una **lista de particiones libres**
- Variantes
 - Cola única de procesos
 - Múltiples colas de procesos
- Problema:
 - **Fragmentación interna**



Asignación contigua de memoria

• Particiones variables

- **Inicialmente** la memoria disponible para procesos esta toda libre en un **único hueco**
- Conforme llegan demandas de procesos se va asignando memoria
- El sistema operativo mantiene una **lista de huecos que contiene su dirección de inicio y su tamaño**



- Problema:
 - **Fragmentación Externa**
- Estrategias de asignación de huecos:
 - **Mejor hueco (Best Fit)**

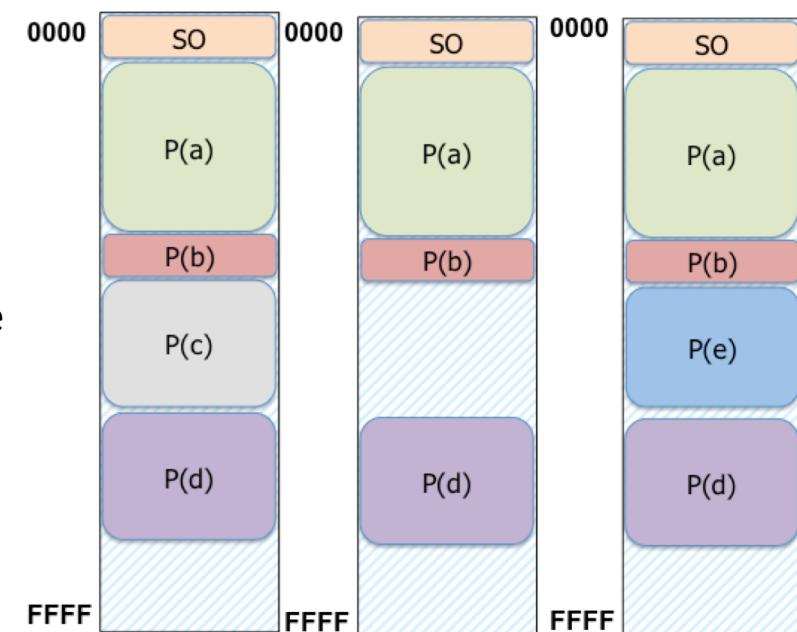
Ubica al proceso en el menor hueco que satisface su petición

- **Peor hueco (Worst Fit)**

Ubica al proceso en el mayor hueco que satisface su petición

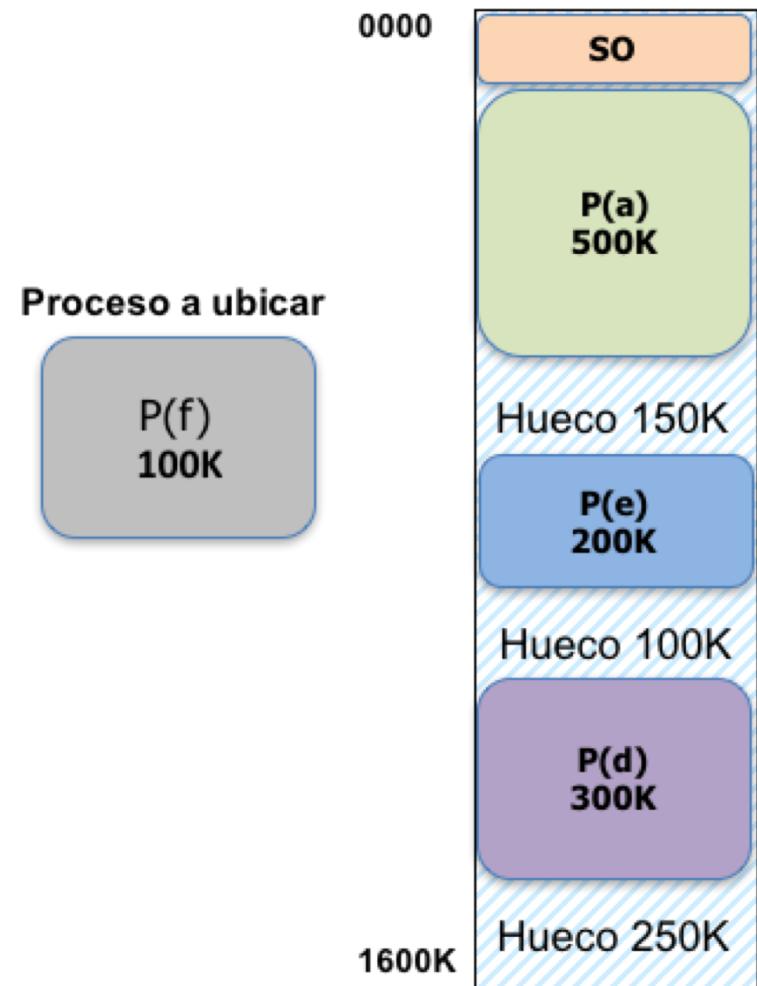
- **Primer hueco (First Fit)**

Ubica al proceso en el primer hueco que satisface su petición



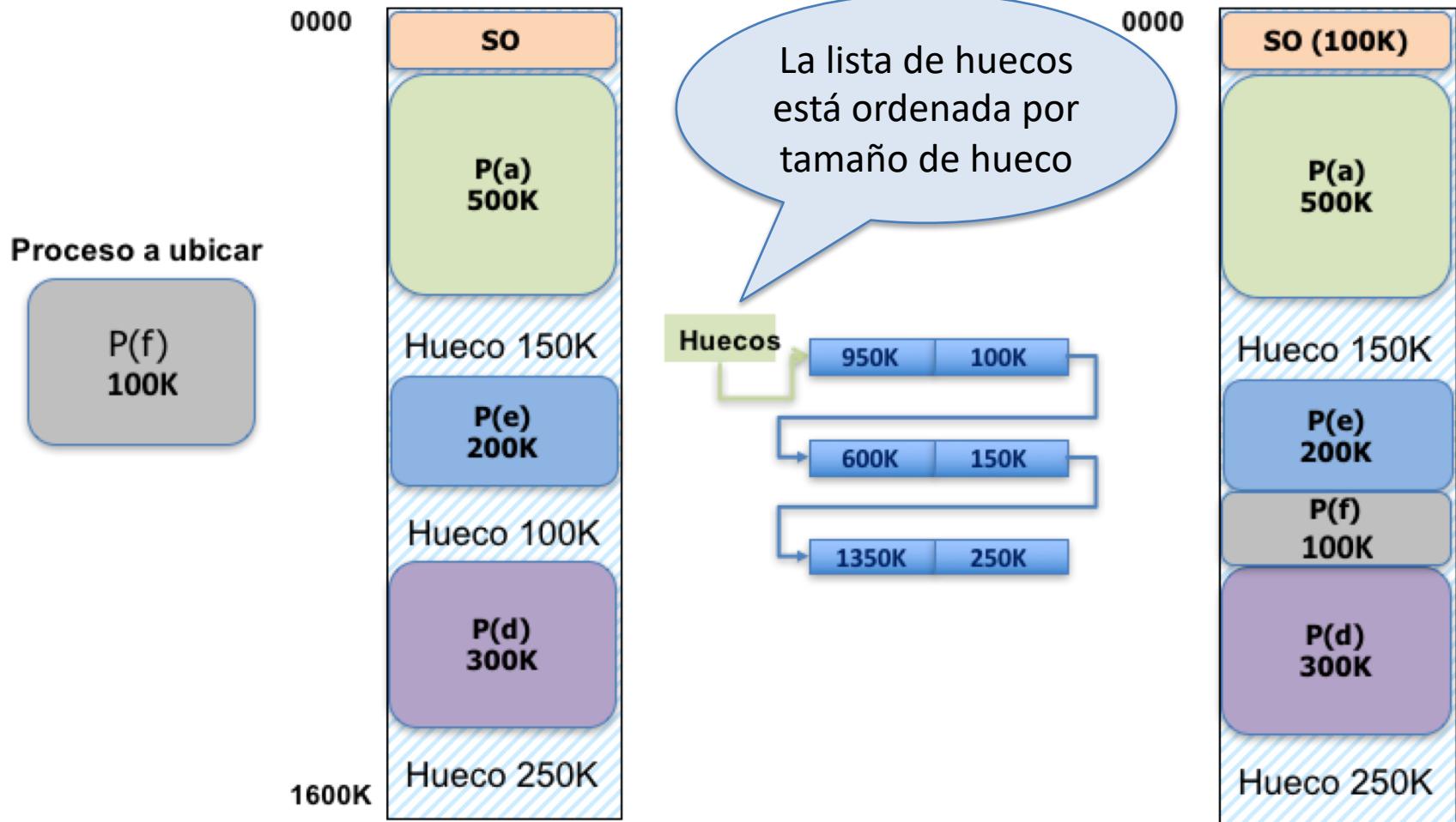
- Particiones Variables

¿Dónde se ubicaría el proceso $P(f)$ en cada una de las estrategias de asignación de huecos?



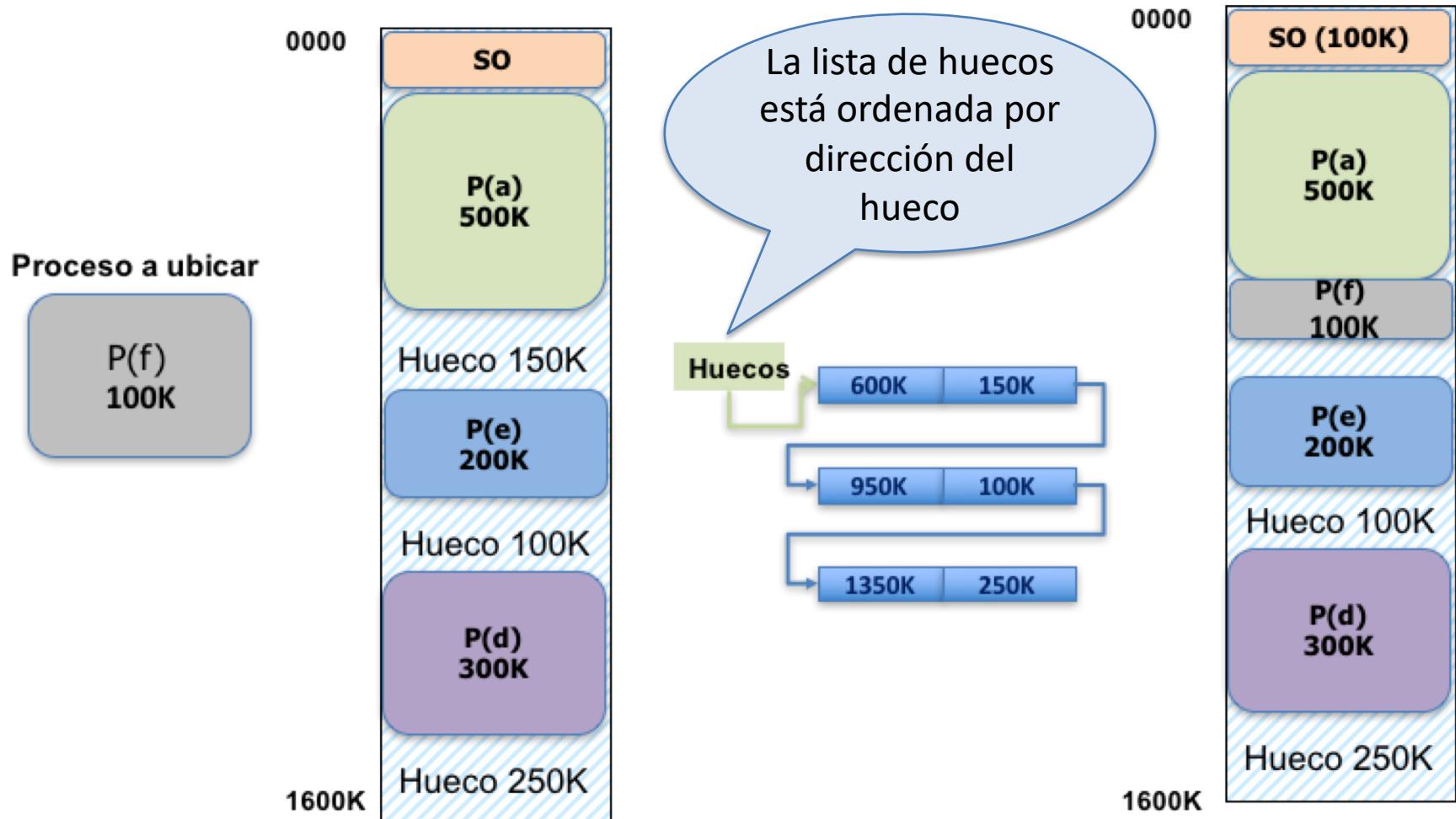
Asignación Contigua

- Estrategia Mejor Ajuste



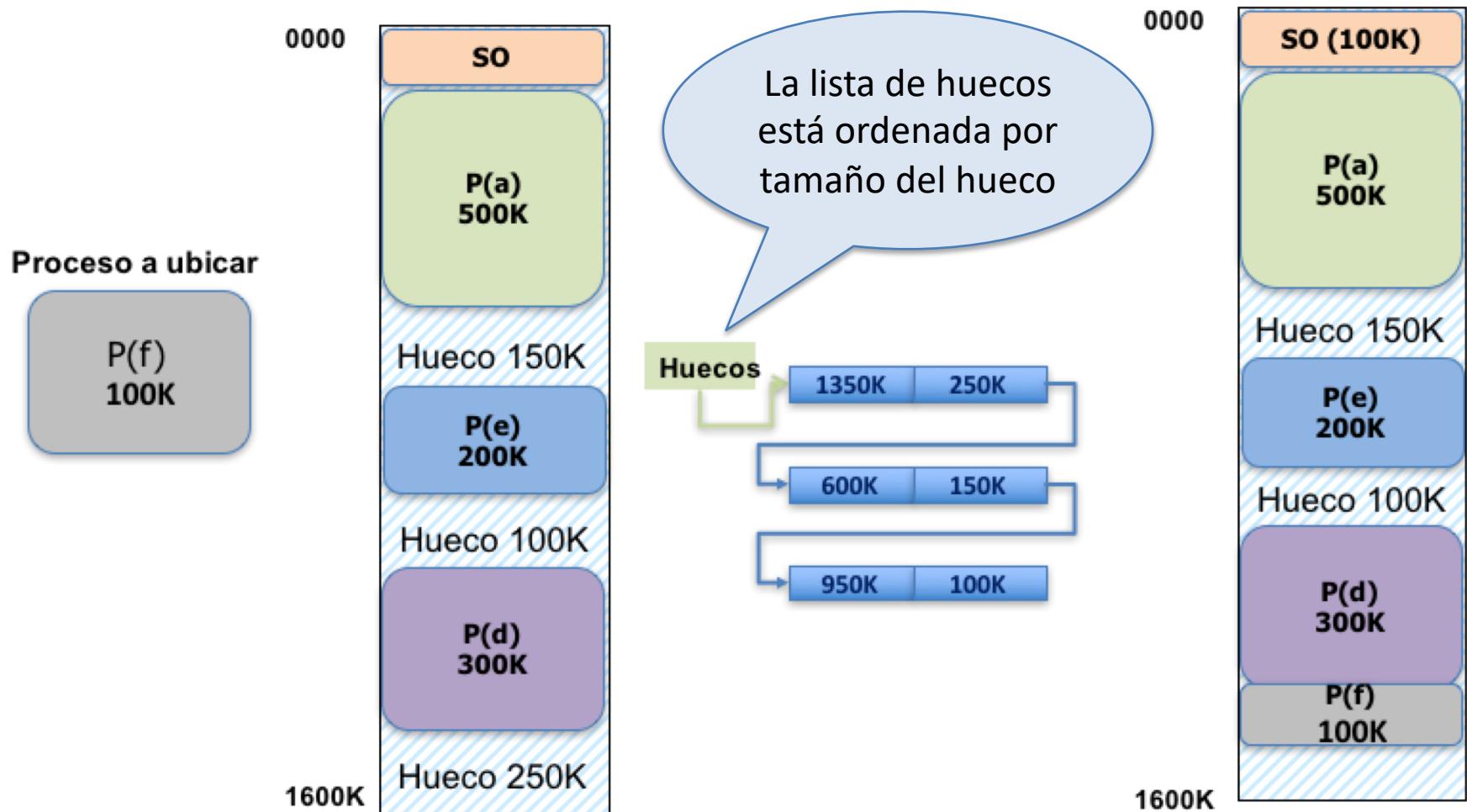
Asignación Contigua

- Estrategia Primer Ajuste



Asignación Contigua

- Estrategia Peor Ajuste

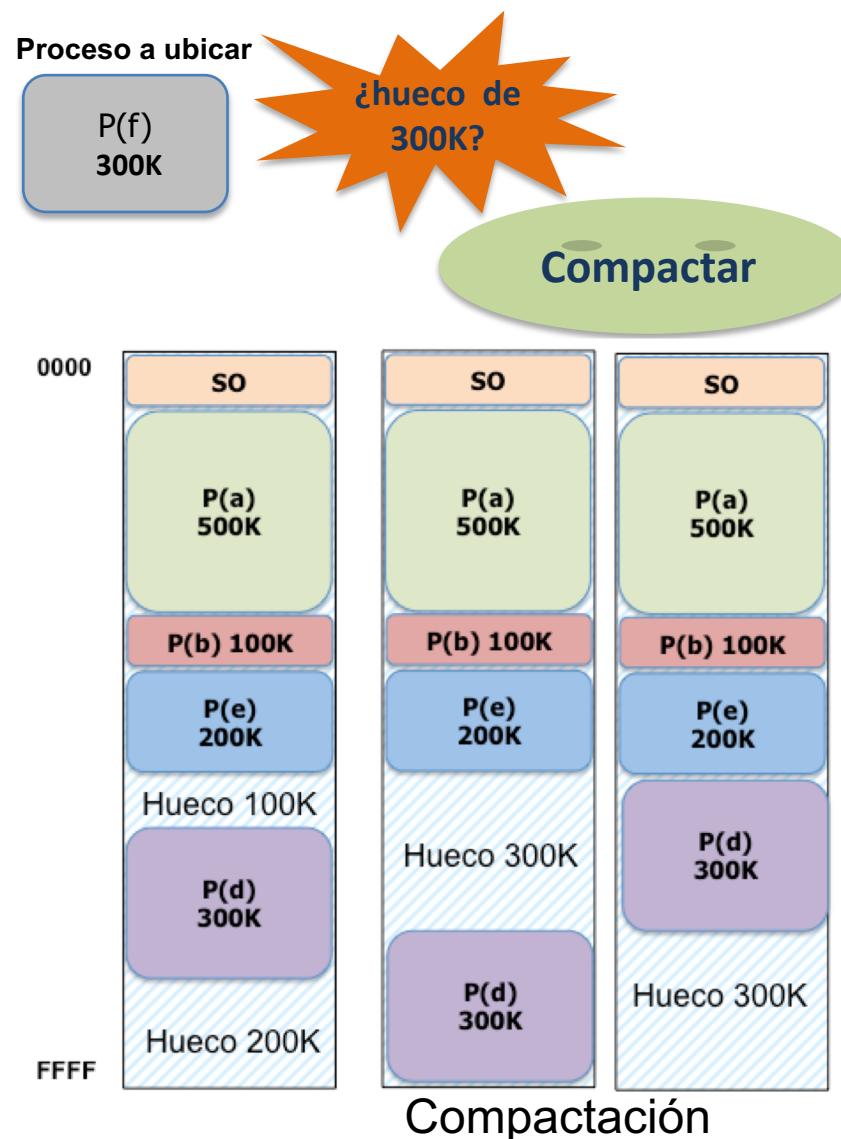


• Fragmentación externa

- Sumando los tamaños de los huecos existentes habría suficiente memoria libre para satisfacer una demanda, pero al no estar contiguos no se puede asignar al proceso

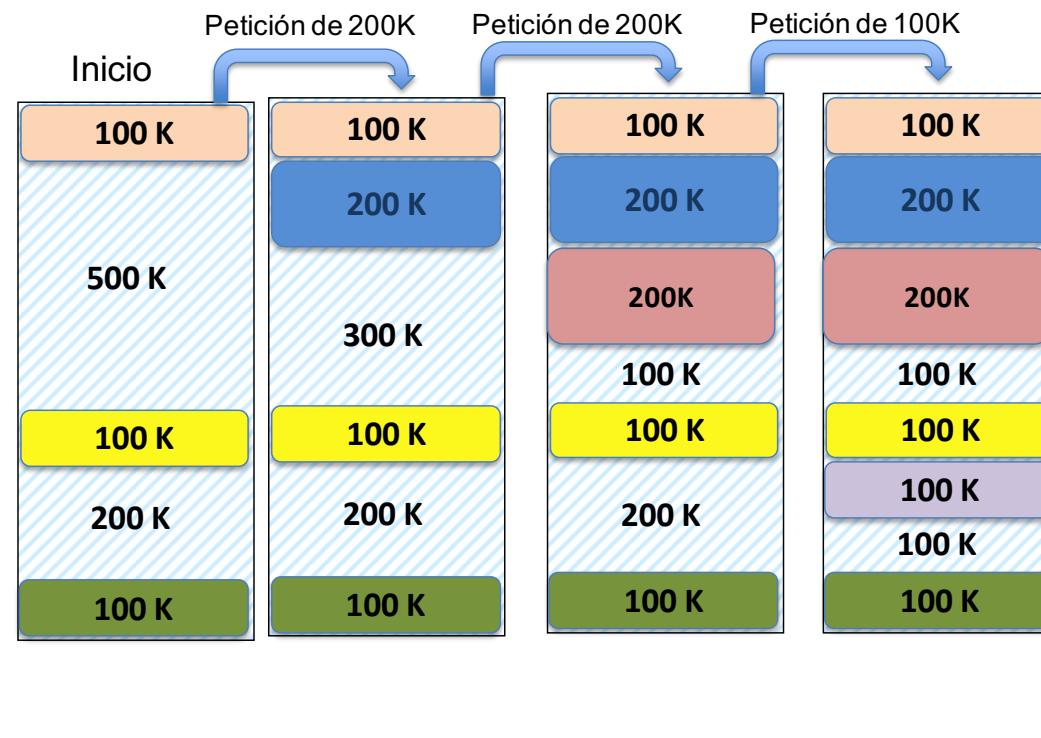
• Compactación

- Solución para la fragmentación externa
- Los procesos se reubican en memoria
- Imprescindible código reubicable en tiempo de ejecución para compactar



Ejercicio 7.1:

- Sea un sistema operativo que gestiona la memoria mediante **asignación contigua con particiones dinámicas (variables)**. Partiendo de un estado de la memoria indicado como Inicio, la ocupación de la misma ante las peticiones realizadas ha sido la que se representa en los gráficos:



- Determine y justifique el algoritmo utilizado entre **mejor ajuste, peor ajuste y primer ajuste** para la asignación de las demandas.

- Un sistema operativo gestiona su memoria de 8196 palabras mediante **asignación contigua con particiones variables**. En un instante t el sistema está ejecutando tres procesos de tamaño:

Proceso A de tamaño 1024 palabras

Proceso B de tamaño 3072 palabras

Proceso C de tamaño 3584 palabras

Razone si los siguientes escenarios independientes entre sí son posibles:

- a) Que el proceso **B** genere la dirección lógica **960** y que ésta sea traducida por la dirección física **725**
- b) Que el proceso **A** genere la dirección lógica **1500** y que ésta sea traducida por la dirección física **1500**
- c) Que el proceso **C** en el instante t genere la dirección lógica **525**, que se traduzca por la dirección física **2061** y en el instante **t+10** la misma dirección lógica se traduzca por la dirección física **1549**

- Memoria de asignación contigua
 - Objetos interactivos para aprendizaje
 - [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 cola.htm](http://labvirtual.webs.upv.es/Fijas_Una_colा.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)