

Tema 2: Gestión de Procesos

1. Introducción
2. Conceptos fundamentales
3. Implementación de procesos
4. Hilos (Threads)
5. Planificación de la CPU

Bibliografía

- W. Stalling. **Sistemas Operativos**. 5ª Edición.
Capítulos 3, 9 y 10.

Introducción


- Procesamiento concurrente:
 - ▶ Multiprogramación
 - ▶ Multiprocesamiento
 - ▶ Procesamiento distribuido
- Proceso: *Programa en ejecución*
- Proceso \neq programa
- Servicios del SO
 - ▶ Ejecución concurrente
 - ▶ Sincronización de procesos
 - ▶ Comunicación entre procesos

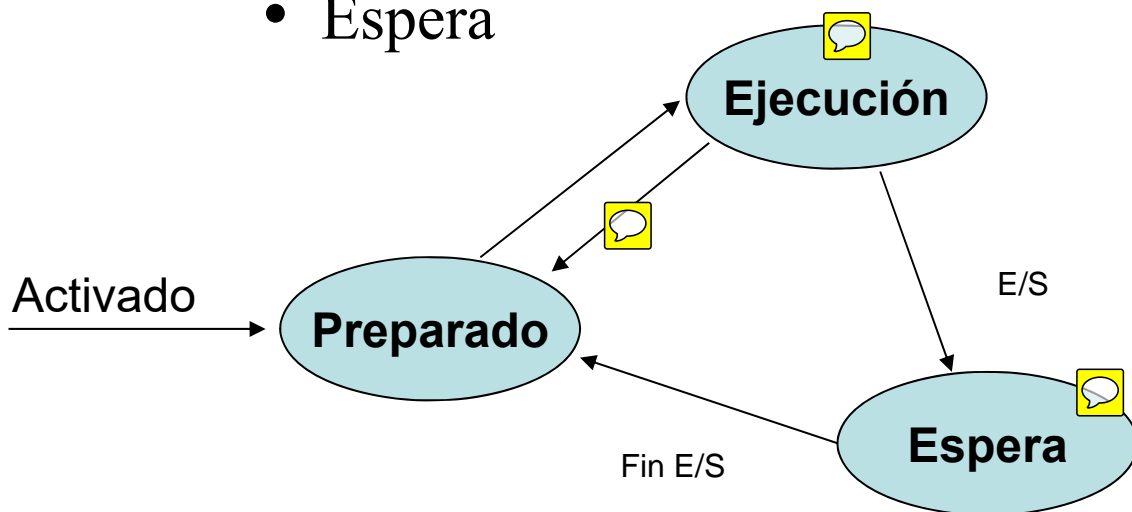
Conceptos fundamentales (I)

- Relación entre procesos:

- ▶ Independientes
- ▶ Cooperativos
- ▶ Competitivos

- Estado de un proceso:

- ▶ Suspendido
- ▶ Activado
 - Ejecución
 - Preparado 
 - Espera



- Estado global del sistema

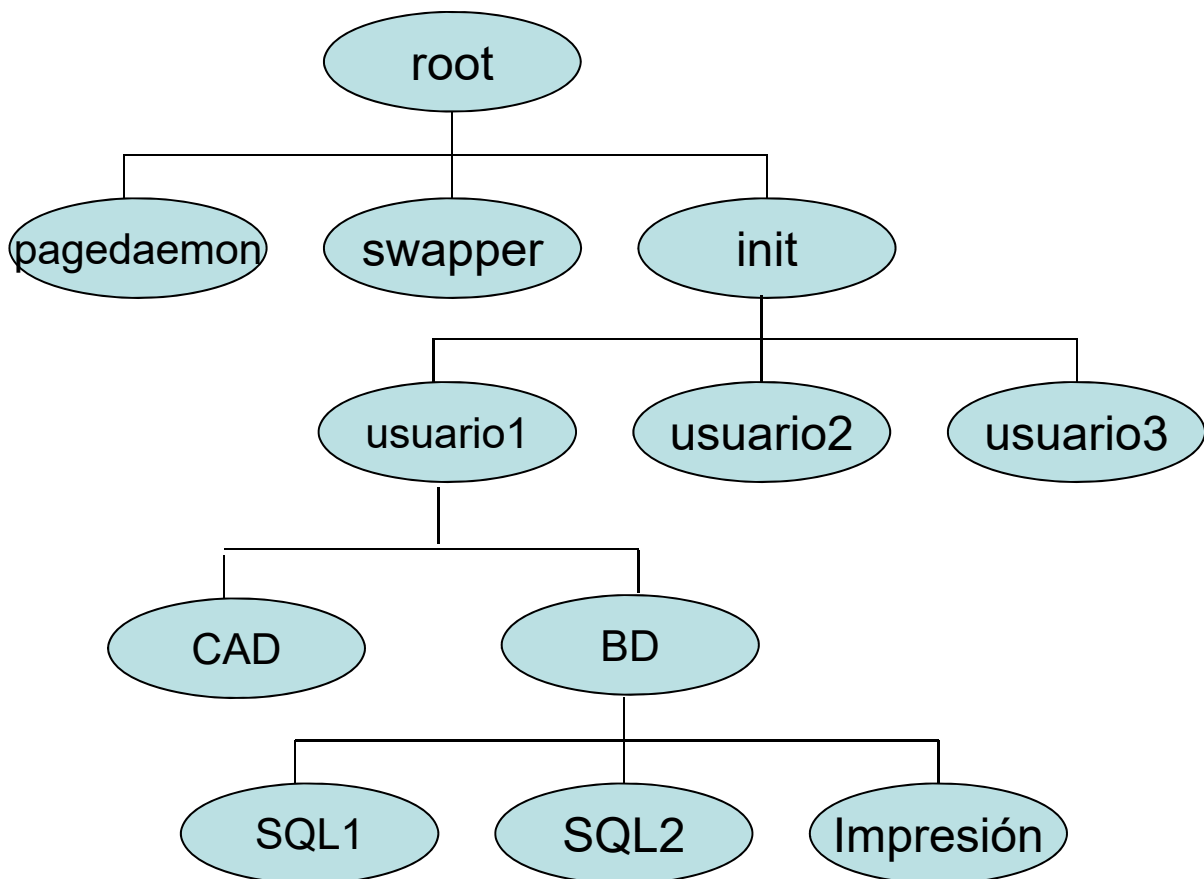
Conceptos fundamentales (y II)

- Operaciones con procesos:

- ▶ Creación de procesos: *fork* 🗨️

- Proceso padre e hijos
 - Árbol de procesos

- ▶ Terminación de procesos: *exit*, *kill* 🗨️ 🗨️




Árbol de procesos en un sistema UNIX representativo

Implementación de procesos (I)

*El **bloque de control de procesos (PCB)** es la estructura de datos del sistema que mantiene toda la información sobre un proceso.*



- El PCB de cada proceso se crea cuando se crea un proceso y se destruye cuando el proceso termina.
- Información que contiene:
 - ▶ Identificador del proceso: *pid*
 - ▶ Estado de un proceso
 - ▶ Registros de la CPU: *CP, flags, ...* 
 - ▶ Información de planificación de la CPU: *prioridad, ...*
 - ▶ Información de gestión de memoria: *punteros, tablas, registros, ...*
 - ▶ Información contable: *tiempo consumido, límites de tiempo, ...*
 - ▶ Información de estado de E/S: *lista de dispositivos de E/S, lista de archivos abiertos, ...*

Implementación de procesos (II)

| |
|--|
| pid |
| estado |
| registros |
| prioridad |
| punteros a memoria |
| lista de archivos abiertos |
| lista de dispositivos de E/S asignados |
| ... |

PCB de un proceso

Implementación de procesos (III)

● Cambio de contexto de un proceso

Retirar de la CPU el proceso en ejecución y asignarla a un proceso en estado de preparado.

► Acciones necesarias

- Salvar el contenido del proceso en ejecución a su PCB
- Restaurar el contexto del nuevo proceso desde su PCB

► Reducen la utilización de la CPU



$$\text{Utilización} = \frac{T(P1) + T(P2) + T(P3)}{T(P1) + T(P2) + T(P3) + 3 \cdot t}$$

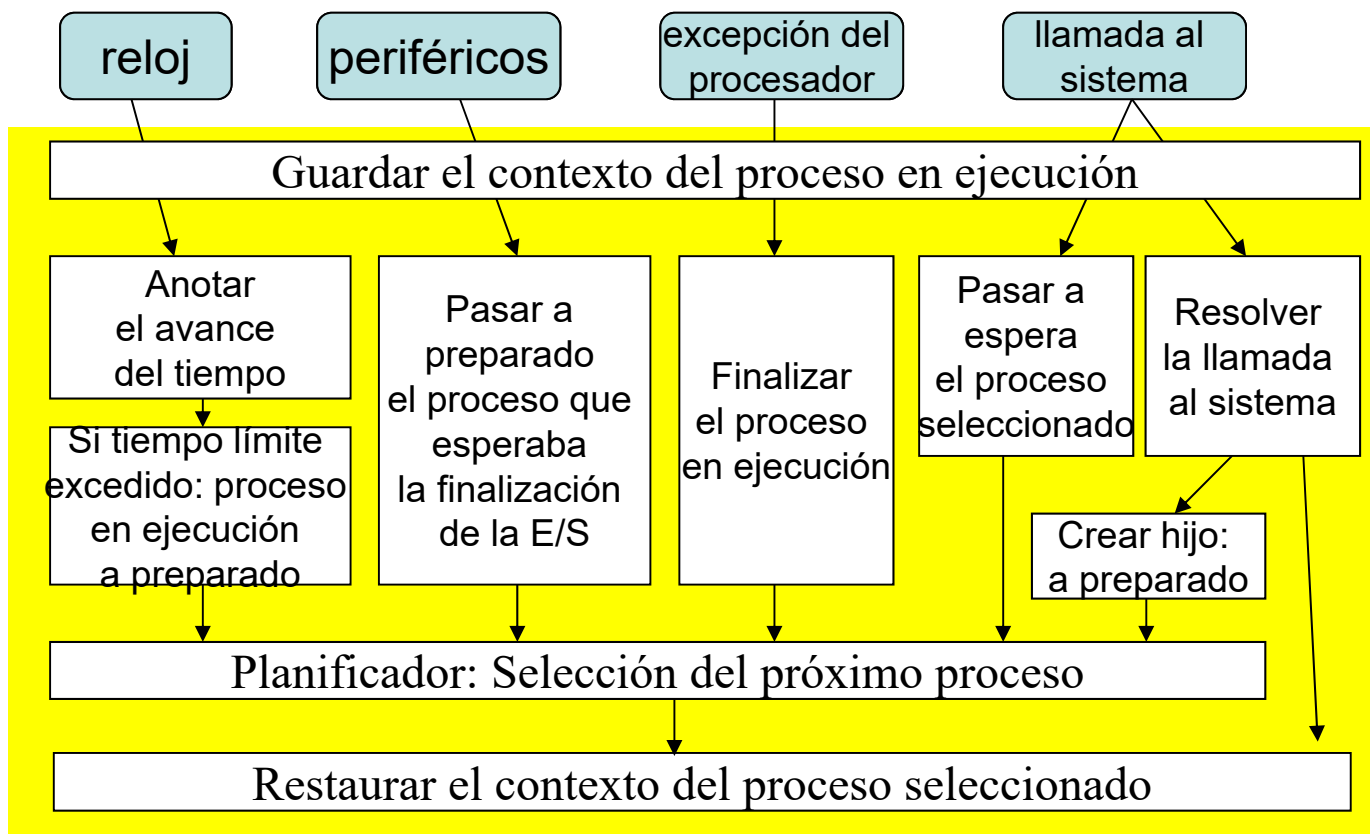
Implementación de procesos (IV)

● Cambio de contexto de un proceso

► Motivos que lo provocan

- Finalización normal de un proceso en ejecución
- Espera de un proceso al realizar una operación de E/S
- Existencia de una interrupción

● SO como manejador de interrupciones

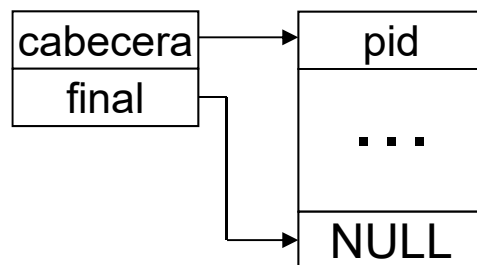


Implementación de procesos (V)

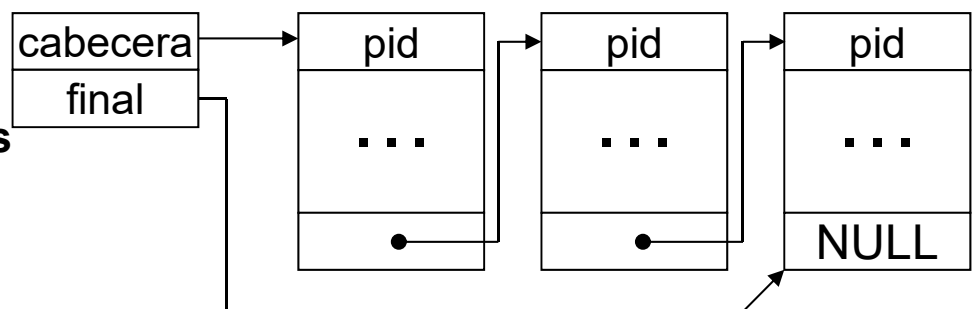
- Colas de procesos: *Tipo Abstracto de Datos que mantiene los PCB de los procesos en una estructura dinámica de lista.*

- ▶ Se mantiene una cola de procesos por cada estado de los procesos.

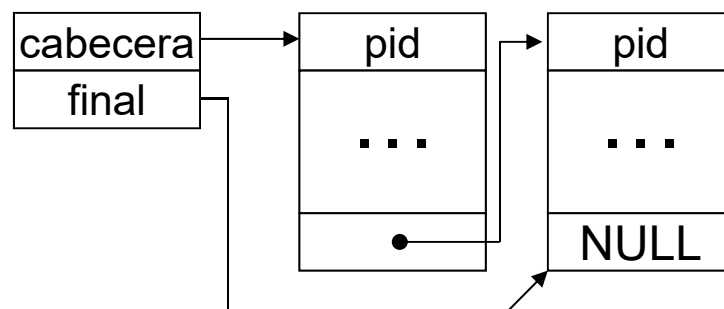
Cola de procesos activos



Cola de procesos preparados



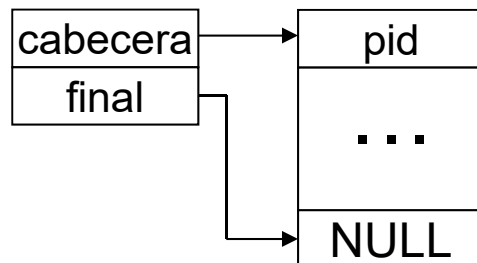
Cola de procesos en espera



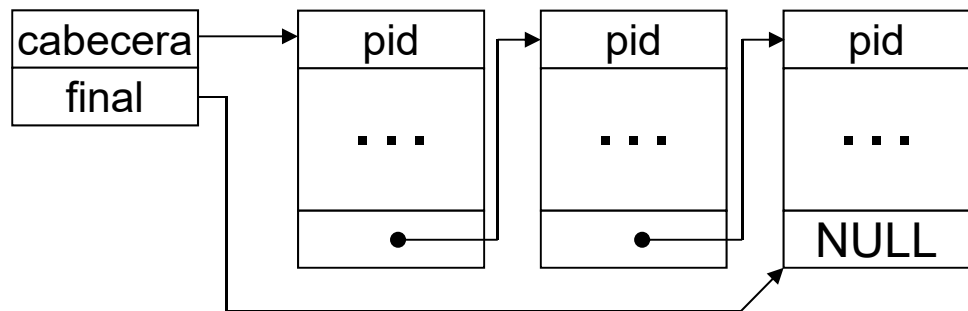
- ▶ Sistemas con varios procesadores
- ▶ Lista de espera por recursos

Implementación de procesos (y VI)

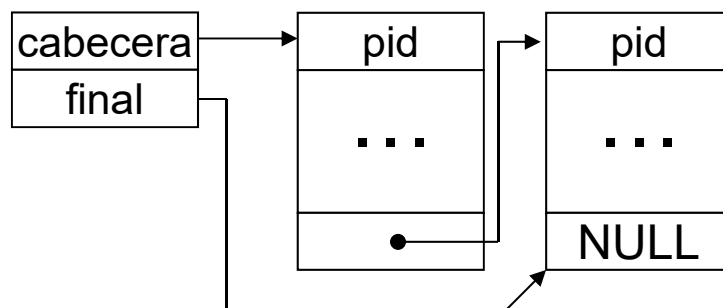
**Cola de procesos
Impresora1**



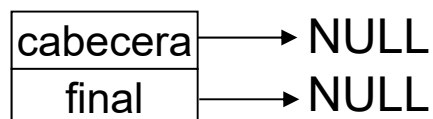
**Cola de procesos
Impresora2**



**Cola de procesos
Fax**



**Cola de procesos
BD**



Hilos (Threads)

Flujos de control independientes dentro de un mismo proceso con pila, variables locales y CP propios.

- Ventajas

- ▶ Posibilidad de compartir un espacio de direcciones y recursos
- ▶ El tiempo empleado en el cambio de contexto es menor que en los *procesos completos*

- Inconvenientes

- ▶ Sincronización entre los hilos y entre los procesos.
- ▶ Complejidad en la programación y depuración

- Diseño de programas basados en hilos

- ▶ Aplicaciones con paralelismo potencial
- ▶ Existencia de numerosas tareas de E/S
- ▶ Existencia de eventos asíncronos

Planificación de la CPU (I)

- Conceptos de planificación
 - ▶ Escasez de recursos
 - ▶ Recursos de acceso exclusivo: CPU, Impresoras, etc.
 - ▶ Política de Asignación
- Situaciones de planificación
 - ▶ Cuando un proceso pasa del estado de ejecución a espera
 - ▶ Cuando un proceso pasa del estado de ejecución a preparado
 - ▶ Cuando un proceso pasa del estado de espera al estado de preparado
 - ▶ Cuando un proceso termina
- Tipos de planificación
 - ▶ Expropiativa
 - ▶ No expropiativa

Planificación de la CPU (II)

- Planificador de la CPU: *Scheduler*
- Cargador: *Dispatcher*
 - ▶ Cambiar de contexto
 - ▶ Saltar al punto apropiado del proceso para continuar desde ahí.
- Criterios de planificación
 - ▶ Utilización de la CPU $\text{Utilización} = \frac{t_{\text{CPU_ocupada}}}{t_{\text{CPU_total}}}$
 - ▶ Rendimiento $\text{Rendimiento} = \frac{\text{nº procesos terminados}}{t}$
 - ▶ Tiempo de retorno
 - ▶ Tiempo de espera
 - ▶ Tiempo de respuesta

Planificación de la CPU (II)

- Algoritmos de planificación no expropiativos
 - ▶ Servicio por orden de llegada (FCFS)
 - ▶ Planificación con prioridad
 - Estática
 - Dinámica
 - ▶ Primero la tarea más corta (SJF)

Planificación de la CPU (III)

- Algoritmos de planificación expropiativos
 - ▶ Planificación por turno circular (RR)
 - ▶ Planificación con prioridad
 - ▶ Tiempo que queda más corto (SRT)
 - ▶ Planificación con colas multinivel (MLQ)
 - Número de colas
 - Algoritmo de planificación para cada cola
 - Prioridad de las colas
 - Criterio que indique la cola en la que entra una tarea cuando necesite un servicio
 - ▶ Planificación con colas multinivel con realimentación (MLFQ)
 - Criterio que determine el movimiento de las tareas entre las colas
- Planificación de múltiples procesadores
- Planificación en tiempo real

Planificación de la CPU (y IV)

- Evaluación de algoritmos

- ▶ Criterios

- Maximizar la utilización
 - Maximizar el rendimiento
 - Otros o combinación de ellos.

- ▶ Modelo determinista

- ▶ Modelo de colas

- ▶ Simulación

- ▶ Implementación