

Multiprocesamiento en Lenguaje C

Creación y manejo de procesos en C

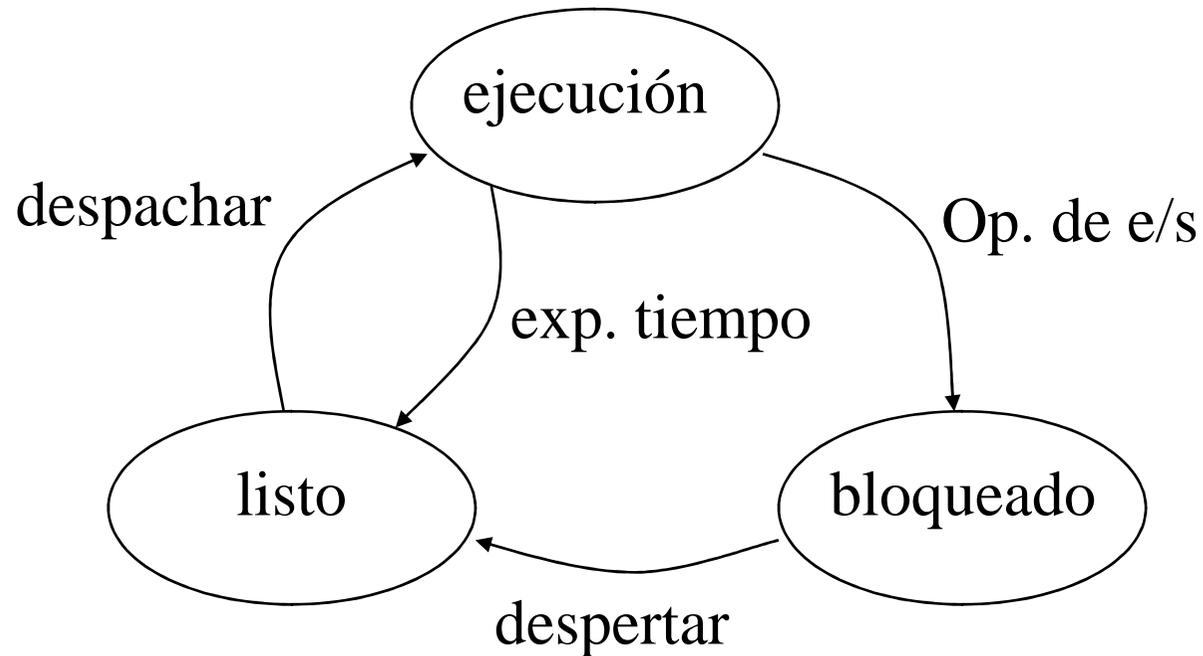
Proyecto PAPIME PE104911

Pertinencia de la enseñanza del cómputo paralelo en el
currículo de las ingenierías

Creación y manejo de procesos en C

Conceptos básicos

- **Estados de un proceso**



Creación y manejo de procesos en C

Estructura de un proceso

- Los procesos están compuestos por tres bloques o segmentos:
 - ⊗ Segmento de texto. Código ejecutable del programa.
 - ⊗ Segmento de datos. Variables globales y estáticas.
 - ⊗ Segmento de pila (stack). Dinámico, llamadas a funciones o al sistema:
 - modo usuario
 - modo kernel

Creación y manejo de procesos en C

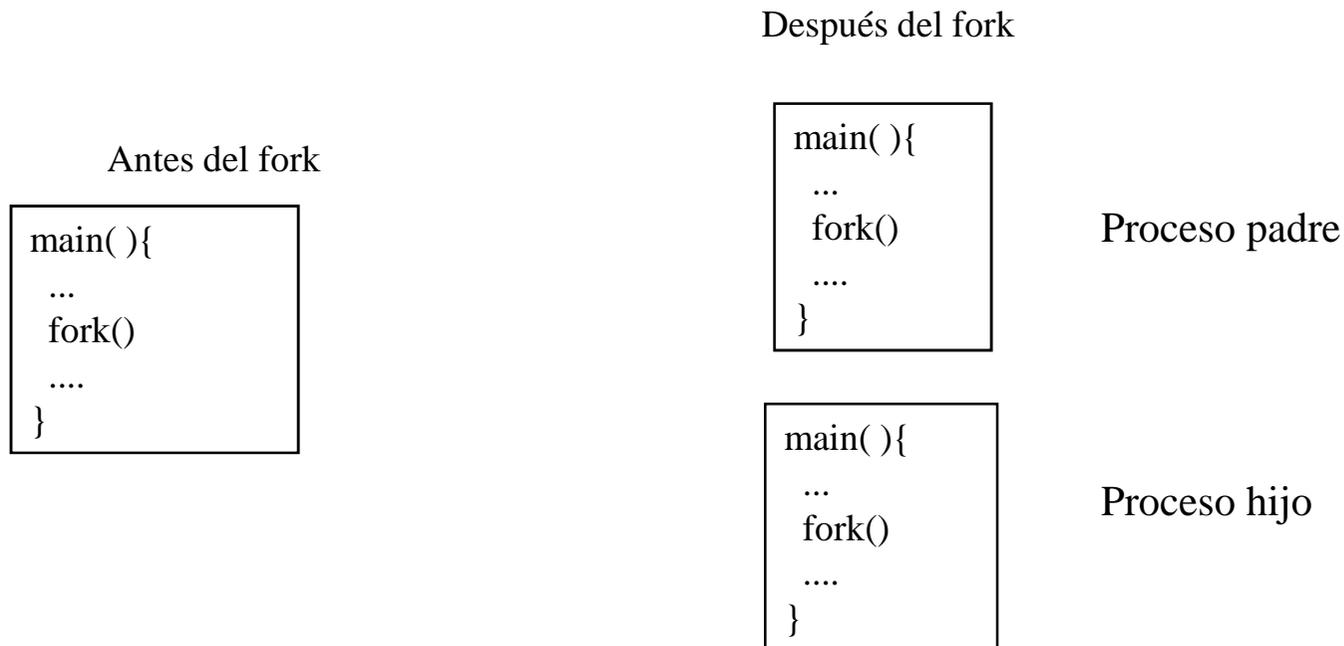
Estructura de un proceso

- Se utiliza una *Tabla de procesos*, requerido para el control de éstos.
 - * Estado
 - * Apuntadores a memoria ocupada
 - * UID
 - * PID
 - * Descriptores de eventos
 - * Parámetros de planificación
 - * Campo de señales
 - * Temporizadores para cálculo de prioridad dinámica

Creación y manejo de procesos en C

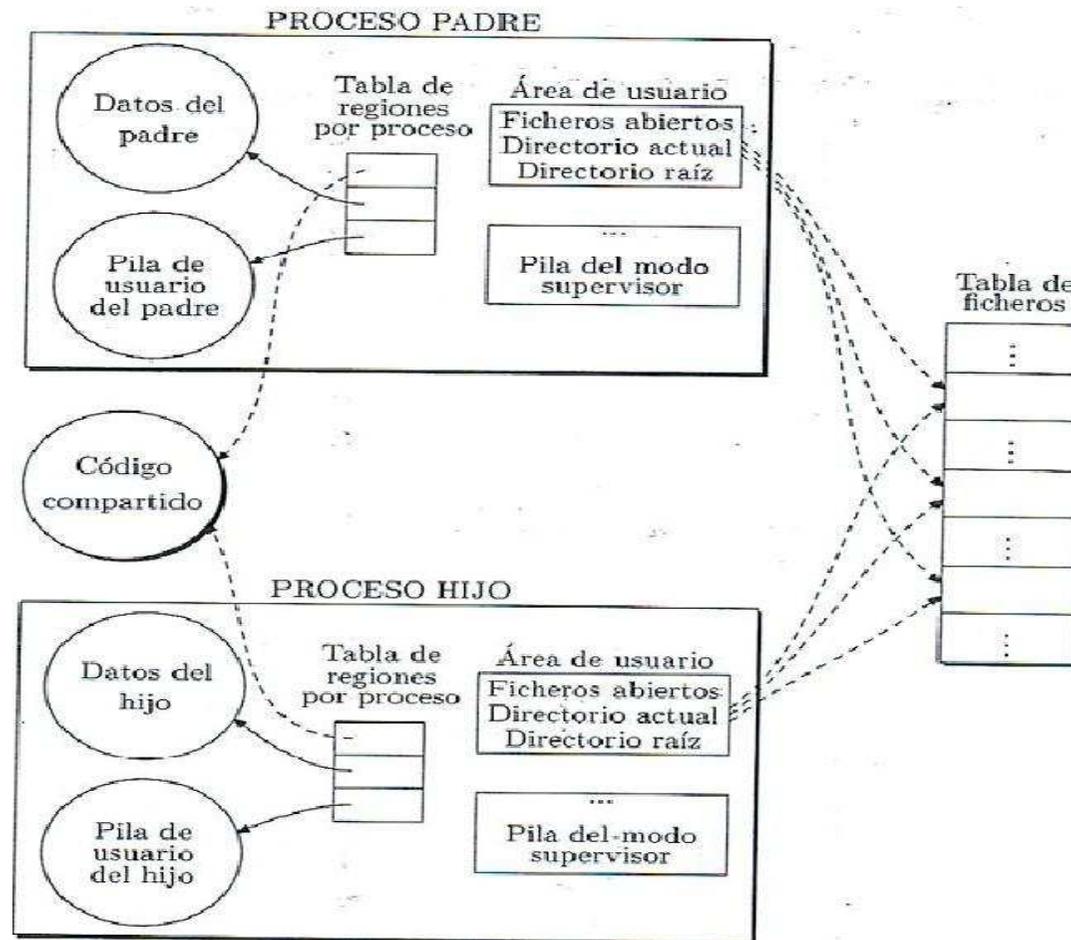
Creación de un proceso

- Todos los procesos, excepto el primero, son creados mediante la llamada al sistema *fork()*



Creación y manejo de procesos en C

Creación de un proceso con fork()



Creación y manejo de procesos en C

Manejo de procesos en UNIX

Se realiza a través de llamadas al sistema.

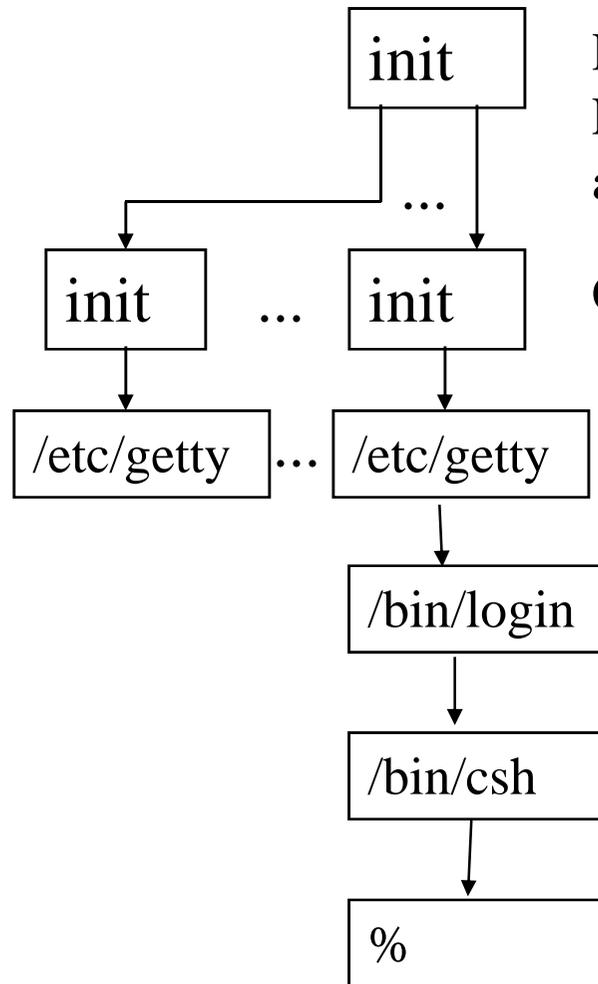
Ejemplos:

- Ejecutar un programa: familia *exec*.
- Detener un proceso: *wait*
- Terminar: *exit*
- Comunicación entre procesos: *pipe* y *sockets*

Creación y manejo de procesos en C

Aplicación de manejo de procesos

Shell en Unix



Proceso creado por el kernel al iniciar Unix
Lee /etc/getty para saber qué terminales serán activadas y hace *fork*

Cada uno de los init hijos hace exec hacia /etc/getty

Pone “login:” y espera el nombre del usuario.
Cuando recibe el nombre hace *exec* a /bin/login

Mira al archivo /etc/passwd para conocer el shell a utilizar

/bin/login hace exec hacia el shell correspondiente

/bin/csh pone el prompt y espera comando de usuario