

UD1. Programación multiproceso





Multitarea

- Consiste en la ejecución simultánea de más de un proceso en un procesador a lo largo de un intervalo de tiempo
- Rápida alternancia entre procesos
 - Genera al usuario la ilusión de que se están ejecutando simultáneamente
 - Realmente un único proceso en ejecución en cada momento
- Alternancia entre procesos mediante cambio de contexto
- Compensa la multitarea con un procesador?
 - Puede parecer que no porque el tiempo de ejecución de los procesos lo hay que sumar al tiempo necesario para los cambios de contexto y para la planificación
 - ...pero sí porque es poco tiempo frente a las ventajas de ejecutar varios procesos al mismo tiempo
 - Interactividad realizando varias tareas a la vez
 - Mayor aprovechamiento del uso de la CPU



Sistemas monoprocesadores y multiprocesadores

- sistema monoprocesador: tiene un único procesador
- sistema multiprocesador: tiene varios
- en ambos se pueden ejecutar varios procesos de manera simultánea:
 - mediante técnicas de programación concurrente que permiten la comunicación y sincronización entre procesos
- de forma
 - real: en sistemas multiprocesadores
 - simulada: en sistemas monoprocesadores (y también multi)



Sistemas monoprocesadores

- La ejecución concurrente de varios procesos en un sistema monoprocesador se conoce con el nombre de multiprogramación
- El cambio de contexto es el mecanismo
 - que permite la ejecución de un nuevo proceso en la memoria
 - que incluye la salvaguarda previa del estado de ejecución del proceso actual.



Sistemas multiprocesadores

- La programación paralela
 - consiste en la ejecución de varios procesos concurrentes en un sistema multiprocesador
- dependiendo de su arquitectura
 - Sistemas fuertemente acoplados
 - existe una memoria compartida para todos los procesadores
 - todos acceden a la misma y el sistema de E/S a través del mismo bus de conexión
 - los hay simétricos (todos iguales) y asimétricos (el maestro controla los esclavos)
 - Sistemas débilmente acoplados
 - no hay memoria compartida
 - cada procesador tiene su propia memoria y su propio sistema de E/S
 - existe una red de comunicaciones entre los procesadores
 - Ejemplos son los clúster o los sistemas distribuidos



Programas y procesos

- Programa
 - objeto estático
 - normalmente almacenado en un fichero binario
 - En un medio de almacenamiento secundario
- Proceso
 - una instancia de un programa en ejecución
 - entidad dinámica
- La ejecución de un programa comienza con la creación y ejecución de un proceso



Ejecución. PCB y Jerarquía

- Para ejecutar un programa
 - primero debe cargarse en la memoria
 - puede utilizar distintos recursos del sistema
 - Memoria principal
 - Procesador o CPU
 - Dispositivos de Entrada/Salida
- El PCB (Process Control Block)
 - tiene toda la información asociada a un proceso
 - Cada proceso tiene su PCB
- Jerarquía de procesos: de un proceso se pueden crear otros (fork)



Clasificación de los tipos de procesos

- Por lotes: formados por una serie de tareas
 - resolución de un algoritmo matemático, escaneo de antivirus, etc.
- Interactivos: interacción continua con el usuario
 - navegador, procesador de textos, etc.
- Tiempo real: el tiempo de respuesta es crítico
 - GPS, brazo de robot mecánico, etc.



Administrar procesos: Linux

- top: muestra los procesos en ejecución
- ps: visualiza el estado de los procesos
- kill: envía señal a un proceso
- nice (e renice): cambia la prioridad de un proceso



Administrar procesos: Windows

- Con interfaz gráfica
 - Process explorer
 - Process monitor
 - Taskmgr (Administrador de tareas)
- Por línea de comandos
 - Tasklist
 - Get-Process (y otros cmdlets de powershell)



Hilos

- Para crear un proceso
 - es preciso reservar espacio en la memoria para él
 - requiere mucho tiempo y recursos del sistema
- La creación de un nuevo hilo para un proceso existente
 - Se hace de manera sencilla y rápida
 - no requiere reservar e inicializar espacio en la memoria porque ésta es compartida por todos los hilos de un mismo proceso
 - Se precisan mecanismos de sincronización entre hilos para evitar problemas en el uso de la memoria compartida



Servicios

- son
 - un tipo particular de proceso
 - ejecutados en segundo plano (background)
 - normalmente iniciados por el SO durante su arranque
- pueden proporcionar servicio
 - a otros procesos en el mismo equipo
 - a otros ordenadores
 - Por ejemplo un servidor multihilo
 - puede crear un nuevo hilo para responder a cada nueva petición recibida
 - mantener un pool de hilos, de manera que a cada nueva petición se le puede asignar inmediatamente un hilo ya disponible en el pool



Servicios

- En Windows
 - Gestionables mediante interfaz gráfica con el administrador de servicios (services.msc)
- En Linux
 - Con el comando systemctl
 - Status: muestra el estado
 - start / stop / restart : arranca/para/reinicia el servicio
 - reload: alternativa a restart sin necesidad de reiniciar el servicio
 - enable/disable arranque / no arranque automática



Programación concurrente

- Permite incrementar el rendimiento (+)
- Hace posible un menor tiempo de respuesta (+)
- Un sistema multiprocesador redundante tiene procesadores suplementarios que proporcionan alta disponibilidad y tolerancia a fallos. (+)
- Difícil de implementar mecanismos adecuados de sincronización y comunicación entre procesos (-)
- Sobrecarga, puede minorar las mejoras de rendimiento (-)
- La red puede ser cuello de botella en sistemas distribuidos (-)



Sistemas distribuidos

- Ordenadores independientes comunicados mediante una red
 - Heterogeneidad del hardware
 - independencia entre los sistemas
 - no hay memoria compartida
 - comunicación suele utilizar los protocolos TCP o UDP.
- La programación distribuida consiste en la ejecución de varios procesos concurrentes en un sistema distribuido
- Ventajas
 - son altamente escalables y configurables
- Inconvenientes
 - son sistemas heterogéneos y complejos de mantener
 - la sincronización entre procesos puede ser también compleja
 - la red puede ser de cuello de botella



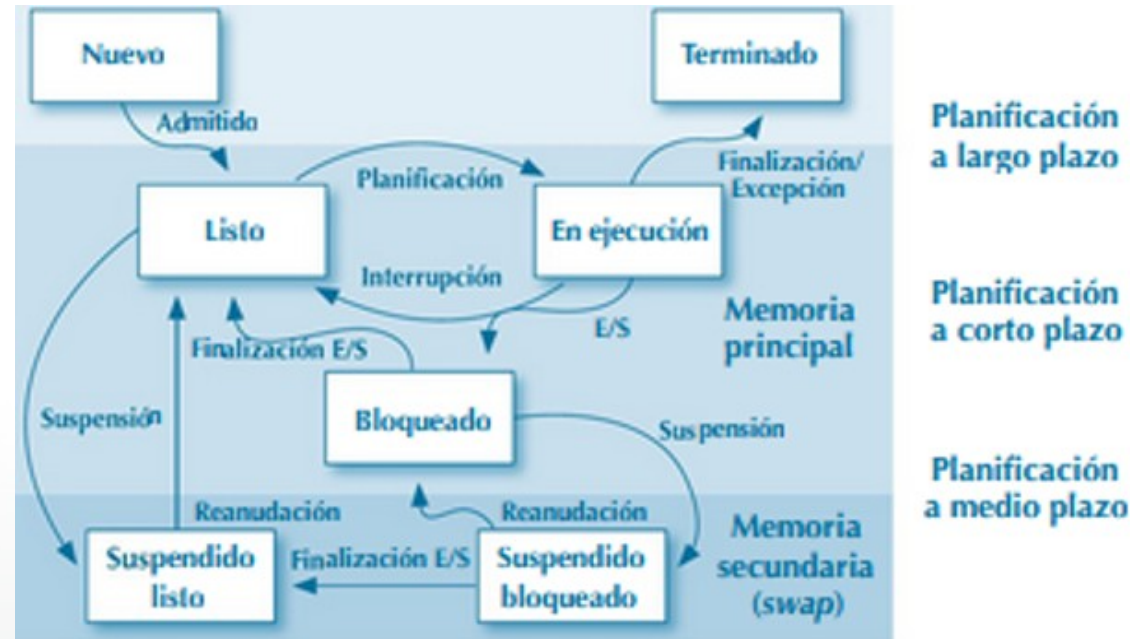
Kernel

- Es la parte central del sistema operativo
- Da respuesta a multitud de eventos mediante un mecanismo de gestión de interrupciones

Estados y planificadores

- El planificador

- a largo plazo decide qué procesos son admitidos para a su ejecución
- a medio plazo gestiona el paso de procesos de la memoria principal a la secundaria (suspensión) y viceversa (reanudación)
- a corto plazo reparte e tiempo de procesador entre todos los procesos





Programación concurrente en Java

- La máquina virtual de Java, el lenguaje y la biblioteca de clases estándar fueron diseñadas para soportar programación concurrente
- La funcionalidad básica para
 - procesos → la clase `Process` y `ProcessBuilder`
 - hilos → la clase `Thread`