

# Planificación de procesos

Gestión de usuarios y procesos

# Introducción

- Los sistemas operativos actuales son multitarea y multiusuario. Esto hace necesario que el sistema operativo sea capaz de trabajar de forma concurrente con varios procesos.
- El sistema operativo se encarga de ir asignando recursos a los diferentes procesos para que se puedan ejecutar simultáneamente.
- Cada una de las ejecuciones de un programa se denomina tarea o proceso.
- Existen comandos que permiten modificar la planificación de ejecución de procesos que lleva a cabo el sistema operativo.

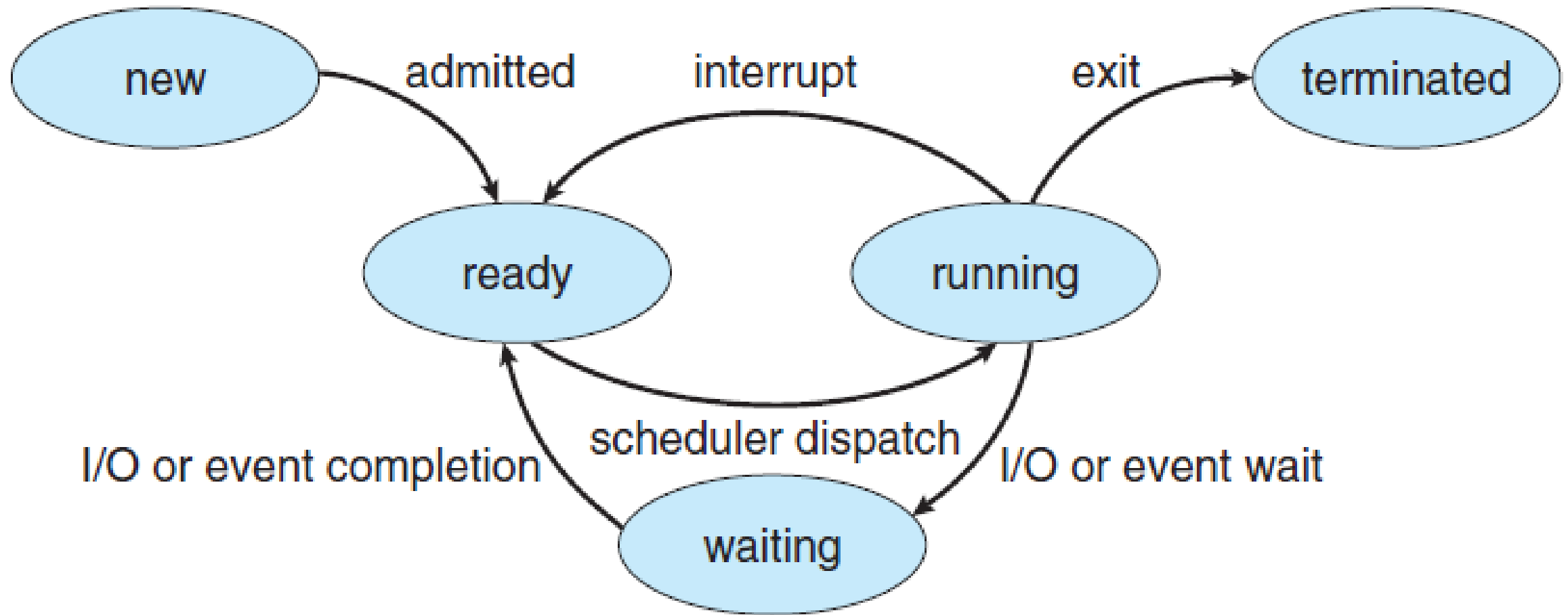
# Introducción

- El sistema operativo gestiona todos los procesos mediante operaciones de creación, comunicación, compartición y finalización de procesos.
- El módulo del sistema operativo encargado de esta gestión se llama **planificador de procesos**.
- El planificador se encarga de establecer el estado de cada proceso y modificarlo atendiendo a un **algoritmo de planificación**.
- Los planificadores convencionales asignan un tiempo de planificación o **quantum** a los procesos en ejecución.

# Proceso de planificación

- El planificador mantiene una cola de prioridad con los procesos que se están ejecutando.
- Se asigna un tiempo de ejecución (quantum) a un proceso, este proceso hace uso de las CPUs disponibles.
- Una vez terminado el tiempo asignado, se produce una interrupción de reloj, que hace que el proceso se detenga y se devuelva el control al sistema operativo.
- A continuación, el sistema operativo toma otro proceso de la cola de prioridad y lo pone en ejecución.
- De esta forma, se consigue que el sistema sea capaz de ejecutar concurrentemente múltiples tareas incluso aunque solo haya una CPU disponible.

# Ciclo de vida de un proceso (lifecycle)



# Ciclo de vida de un proceso (estados)

- **New (nuevo)**: es el primer estado por el que pasa todo proceso, se crean las estructuras de datos para la gestión del proceso y se le asigna un espacio de memoria. La estructura más representativa es el bloque de control de proceso (PCB), donde se almacena la información necesaria para la gestión del mismo.
- **Ready (listo)**: el planificador de procesos establece que el proceso está preparado para pasar a ejecución. Un proceso puede pasar a este estado por:
  - Se acaba de crear (new)
  - Ha vuelto a la cola de listos al resolverse la causa de su espera o bloqueo (waiting)
  - Ha terminado el quantum de tiempo asignado para ejecutarse (running)

# Ciclo de vida de un proceso (estados)

- **Running (ejecutándose):** el proceso está haciendo uso de las CPU disponibles. Es el planificador el que determina qué proceso pasa del estado ready a running, según el algoritmo de planificación.
- **Waiting (esperando o bloqueado):** va a haber procesos en los que, durante su ejecución, haya momentos en los que deben esperar a algo. Este algo puede ser desde una operación de entrada/salida (leer o escribir de una base de datos, fichero...), hasta una señal de otro proceso que estaba haciendo uso de un recurso del sistema necesario la ejecución de este. Cuando un proceso en ejecución se encuentra con esta situación, pasa al estado waiting y libera las CPU para que otros procesos puedan hacer uso de ellas.

# Ciclo de vida de un proceso (estados)

- **Terminated (finalizado)**: el proceso ha terminado de ejecutarse. Se liberan todos los recursos asociados (archivos, buffers y memoria) y se eliminan las estructuras de datos para su gestión (como el PCB del proceso).



# Planificación de procesos

- A priori, no se conoce el tiempo de ejecución de cada proceso, si se va a bloquear y durante cuánto tiempo o su permanencia en la cola de listos. Por tanto, el tiempo de ejecución y el estado de un proceso es impredecible.
- Se denomina cambio de contexto cuando un proceso entra en ejecución mientras otro se retira de ella. Para que esto se produzca se debe modificar la información contenida en las estructuras de datos de gestión del planificador del sistema operativo.
- Cuando se produce un cambio de contexto, toda la información del estado del proceso debe actualizarse en su PCB.

# Modos de ejecución de instrucciones

- **Modo usuario:** es el modo empleado por los programas de usuario y las actividades no críticas del sistema. El sistema puede definir varios modos de usuario, con diferentes prioridades para mejorar el control sobre la planificación de procesos.
- **Modo kernel, núcleo o privilegiado:** es el empleado por los procesos pertenecientes al kernel del sistema operativo para ejecutar las instrucciones. En este modo se puede obtener el control total del procesador. Se ejecutan en este modo tareas como la creación, terminación y sincronización de procesos, gestión y reserva de memoria, la gestión de interrupciones, gestión de accesos al hardware...

# Modos de ejecución de procesos

- **Por lotes o batch:** se lanza un conjunto de tareas para realizar por el sistema y este ejecuta todas ellas sin intervención del usuario. Por ejemplo: la automatización de copias de seguridad, el lanzamiento de una serie de trabajos a la cola de impresión...
- **Interactivo:** un proceso en modo interactivo solicita la interacción del usuario para llevar a cabo su ejecución. El usuario realiza acciones que van produciendo cambios en el sistema. Algunos ejemplos son: un programa de hojas de cálculo, un videojuego, etc...

# Procesos, hilos y tareas en Linux

- **Proceso (process):** en Linux un proceso es una instancia de ejecución de un programa. Por ejemplo, cuando ejecutamos un comando o cuando ejecutamos un programa Java.
- **Hilo (thread):** muchos programas son lo suficientemente complejos como para poder establecer varias secuencias de sentencias independientes (hilos). Por ejemplo, en un programa que gestione peticiones de varios usuarios concurrentes, se podría definir un hilo por cada petición ya que son procesamientos independientes. Gracias a esto, un mismo proceso puede hacer uso de varias CPU simultáneamente.
- **Tarea (task):** el planificador de Linux maneja tareas, las cuales pueden ser o bien procesos de un solo hilo, o hilos de procesos multihilo.

# Ejecución en paralelo vs concurrente

- Cuando varias tareas están siendo ejecutadas de manera simultánea en distintas CPU del sistema, se dice que se están ejecutando en paralelo. Esto es posible en aquellos sistemas que tengan más de una CPU.
- En un sistema informático suele haber múltiples procesos en ejecución, algunos de los cuales pueden hacer uso de varios hilos. Es habitual que no haya suficientes CPU para poder asignar una a cada tarea.
- Es por esto que existe el planificador que se encarga de asignar y desasignar CPU a las tareas de manera que estas puedan ejecutarse de manera concurrente, aunque no necesariamente en paralelo.

# Datos de un proceso

- **PID**: identificador único del proceso
- **PPID**: identifica al proceso que creó este proceso (proceso padre)
- **Usuario** propietario
- Valores del estado del procesador en el momento de producirse el cambio de contexto
- **Estado**
- Valores de referencia de memoria RAM
- Ficheros abiertos
- Buffers de memoria utilizados

# Comando ps

- Gracias al comando ps podemos obtener información de los procesos del sistema.
- Este comando tiene una gran variedad de opciones, para conocerlas detalladamente se debe leer el manual (ejecutando `man ps`)
- Este comando acepta que se pasen opciones siguiendo distintos estándares: UNIX (opciones precedidas por `-`), BSD (opciones sin guión) y GNU (opciones largas precedidas por `--` )

# Opciones típicas del comando ps

- **ps aux**: obtiene información de todos los procesos del sistema: USER, PID, %CPU, %MEM, VSZ, RS, TTY, STAT, START, TIME, CMD
- **ps -ef**: obtiene información de todos los procesos del sistema: UID, PID, PPID, C, STIME, TTY, TIME, CMD.
- **ps axjf**: Imprime información de los procesos en forma de árbol
- **ps -U <usuario> u**: muestra los procesos de un usuario
- **ps --pid <id\_proceso>**: muestra la información del proceso con el id proporcionado.
- **ps -t /dev/pts/<terminal>**: muestra información de los procesos del temrinal virtual seleccionado.



# Salida del comando ps

Columna	Descripción
<b>UID</b>	Id del usuario propietario del proceso
<b>PID</b>	Id del proceso
<b>PPID</b>	Id del proceso padre (proceso que creó este proceso)
<b>C</b>	Índice de utilización del procesador
<b>TTY</b>	Terminal de lanzamiento del proceso
<b>TIME</b>	Tiempo de CPU consumido
<b>COMMAND</b>	Orden de ejecución
<b>VSZ</b>	Tamaño del proceso en la memoria virtual en KB
<b>RSS</b>	Tamaño de la memoria física del proceso en KB
<b>%CPU</b>	Porcentaje de uso de CPU
<b>%MEM</b>	Porcentaje de uso de memoria
<b>STAT o S</b>	Estado del procesador

# Estados del procesador (STAT o S)

Estado	Descripción
R	Ejecutándose o listo para ser ejecutado (Runnable)
S	Bloqueado o durmiendo (Sleeping)
T	Parado (Trace)
Z	Zombie (proceso que ha muerto pero su padre no ha reconocido su muerte y por tanto no ha sido completamente eliminado)
I	Inactivo (Idle)
N	Con prioridad menor de lo normal (NICE)
<	Con prioridad mayor de lo normal
+	Se encuentra en el grupo de procesos en primer plano
s	Proceso líder de sesión
l	Proceso multihilo, estos procesos se componen de varias tareas que se pueden ejecutar en paralelo.