# Procesos

## Concepto de proceso

Los sistemas de procesamiento por lotes ejecutan trabajos, mientras que un sistema de tiempo compartido tiene programas de usuario o tareas. Incluso en un sistema monousuario, como Microsoft Windows, el usuario puede ejecutar varios programas al mismo tiempo: un procesador de textos, un explorador web y un programa de correo electrónico. Incluso aunque el usuario pueda ejecutar sólo un programa cada vez, el sistema operativo puede tener que dar soporte a sus propias actividades internas programadas, como los mecanismos de gestión de la memoria. En muchos aspectos, todas estas actividades son similares, por lo que a todas ellas las denominamos procesos.

En este apunte, los términos trabajo y proceso se usan indistintamente. Aunque personalmente preferimos el término proceso, gran parte de la teoría y terminología de los sistemas operativos se desarrolló durante una época en que la principal actividad de los sistemas operativos era el procesamiento de trabajos por lotes.

## El Proceso

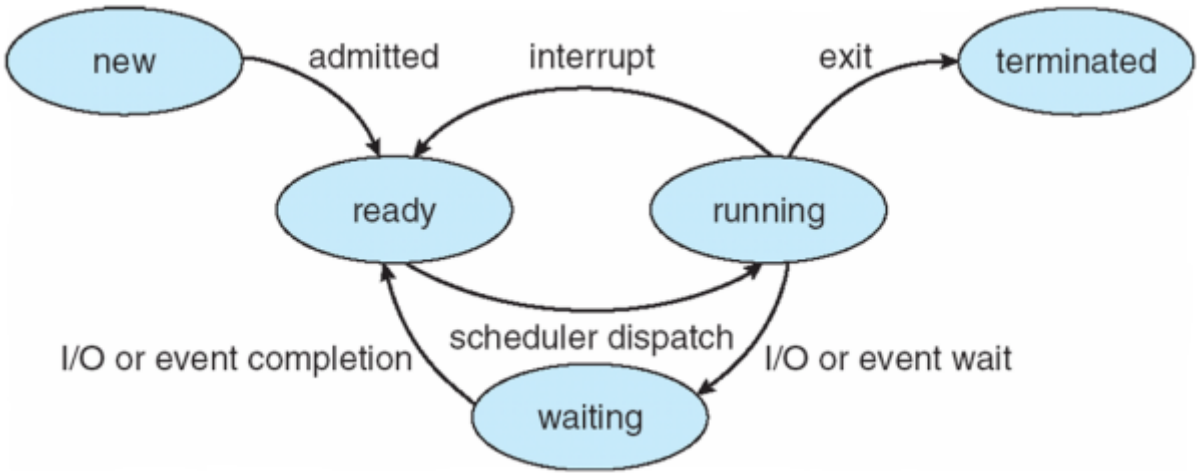
Informalmente, como hemos indicado antes, un proceso es un programa en ejecución. Hay que resaltar que un proceso es algo más que el código de un programa (al que en ocasiones se denomina sección de texto). Además del código, un proceso incluye también la actividad actual, que queda representada por el valor del contador de programa y por los contenidos de los registros del procesador. Generalmente, un proceso incluye también la pila del proceso, que contiene datos temporales (corno los parámetros de las funciones, las direcciones de retorno y las variables locales), y una sección de datos, que contiene las variables globales. El proceso puede incluir, asimismo, un cúmulo de memoria, que es la memoria que se asigna dinámicamente al proceso en tiempo de ejecución.

Insistamos en que un programa, por sí mismo, no es un proceso; un programa es una entidad pasiva, un archivo que contiene una lista de, instrucciones almacenadas en disco (a menudo denominado archivo ejecutable), mientras que un proceso es una entidad activa, con un contador de programa que especifica la siguiente instrucción que hay que ejecutar y un conjunto de recursos asociados. Un programa se convierte en un proceso cuando se carga en memoria un archivo ejecutable. Dos técnicas habituales para cargar archivos ejecutables son: hacer doble clic sobre un icono que represente el archivo ejecutable e introducir el nombre del archivo ejecutable en la línea de comandos (como por ejemplo, prog . exe o a. out.)

Aunque puede haber dos procesos asociados con el mismo programa, esos procesos se consideran dos secuencias de ejecución separadas. Por ejemplo, varios usuarios pueden estar ejecutando copias diferentes del programa de correo, o el mismo usuario puede invocar muchas copias del explorador web. Cada una de estas copias es un proceso distinto y, aunque las secciones de texto sean equivalentes, las secciones de datos, del cúmulo (heap) de memoria y de la pila variarán de unos procesos a otros. También es habitual que un proceso cree muchos otros procesos a medida que se ejecuta.

## Estado del proceso

A medida que se ejecuta un proceso, el proceso va cambiando de estado. El estado de un proceso se define, en parte, según la actividad actual de dicho proceso. Cada proceso puede estar en uno de los estados siguientes:




- **Nuevo**. El proceso está siendo creado.
- **En ejecución**. Se están ejecutando las instrucciones.
- **En espera**. El proceso está esperando a que se produzca un suceso (como la terminación de una operación de E/S o la recepción de una señal).
- **Preparado**. El proceso está a la espera de que le asignen a un procesador.
- **Terminado**. Ha terminado la ejecución del proceso.

Estos nombres son arbitrarios y varían de un sistema operativo a otro. Sin embargo, los estados que representan se encuentran en todos los sistemas. Determinados sistemas operativos definen los estados de los procesos de forma más específica. Es importante darse cuenta de que sólo puede haber un proceso ejecutándose en cualquier procesador en cada instante concreto. Sin embargo, puede haber muchos procesos preparados y en espera.

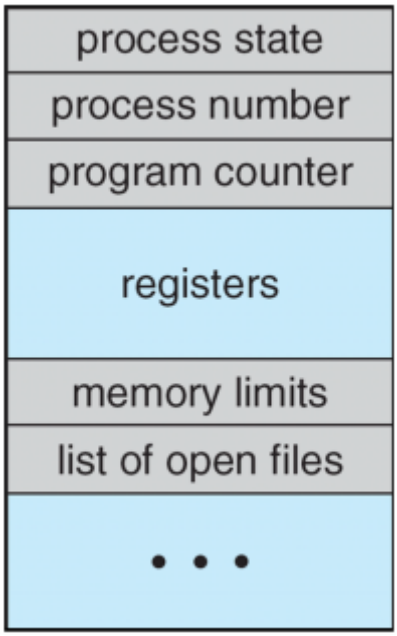
## Bloque de control de proceso

Cada proceso se representa en el sistema operativo mediante un bloque de control de proceso (PCB, process control block), también denominado bloque de control de tarea (véase la Figura 3.3).Un bloque de control de proceso contiene muchos elementos de información asociados con un proceso específico, entre.los que se incluyen:

- **Estado del proceso**. El estado puede ser: nuevo, preparado, en ejecución, en espera, detenido, etc.
- **Contador de programa**. El contador indica la dirección de la siguiente instrucción que va a ejecutar dicho proceso.
- **Registros de la CPU**. Los registros varían en cuanto a número y tipo, dependiendo de la arquitectura de la computadora. Incluyen los acumuladores, registros de índice, punteros de pila y registros de propósito general, además de toda la información de los indicadores de

estado. Esta información de estado debe guardarse junto con el contador de programa cuando se produce una interrupción, para que luego el proceso pueda continuar ejecutándose correctamente.
- **Información de planificación de la CPU**. Esta información incluye la prioridad del proceso, los punteros a las colas de planificación y otros parámetros de planificación que se requieran.
- **Información de gestión de memoria**. Incluye información acerca del valor de los registros base y límite, las tablas de páginas o las tablas de segmentos, dependiendo del mecanismo de gestión de memoria utilizado por el sistema operativo.
- **Información contable**. Esta información incluye la cantidad de CPU y de tiempo real empleados, los limites de tiempo asignados, los números de cuenta, el número de trabajo o de proceso, etc.
- **Información del estado de E/S**. Esta información incluye la lista de los dispositivos de E/S asignados al proceso, una lista de los archivos abiertos, etc.

En resumen, el PCB sirve simplemente como repositorio de cualquier información que pueda variar de un proceso a otro.




## Planificación de procesos

El objetivo de la multiprogramación es tener en ejecución varios procesos al mismo tiempo con el fin de maximizar la utilización de la CPU. El objetivo de los sistemas de tiempo compartido es conmutar la CPU entre los distintos procesos con tanta frecuencia que los usuarios puedan interactuar con cada programa mientras éste se ejecuta. Para conseguir estos objetivos, el planificador de procesos selecciona un proceso disponible (posiblemente de entre un conjunto de varios procesos disponibles) para ejecutar el programa en la CPU. En los sistemas de un solo procesador, nunca habrá más de un proceso en ejecución: si hay más procesos, tendrán que esperar hasta que la CPU esté libre y se pueda asignar a otro proceso.

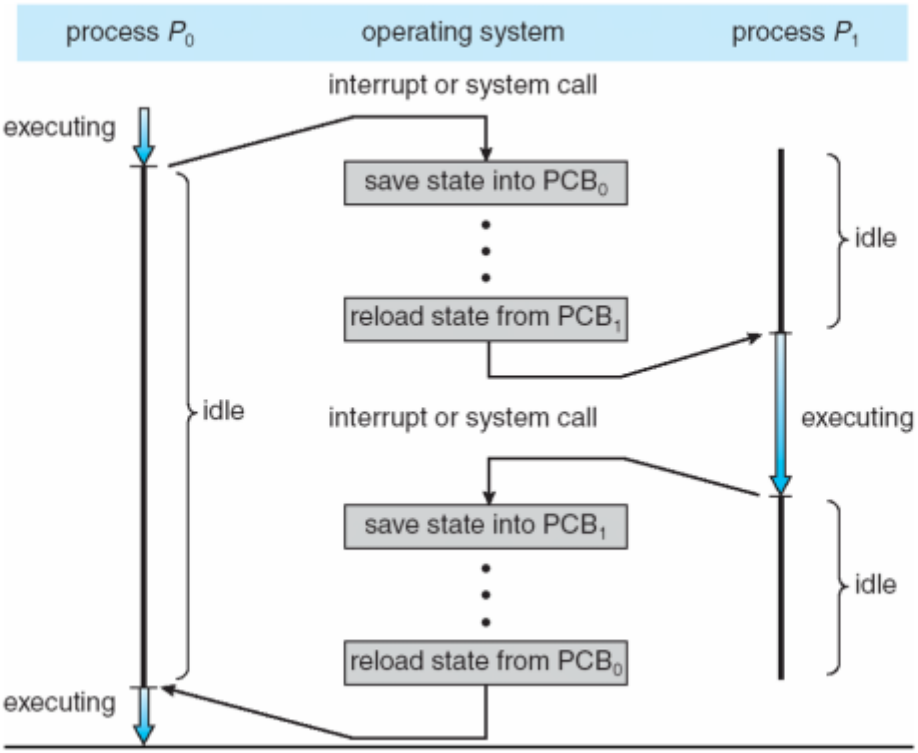
## Colas de planificación

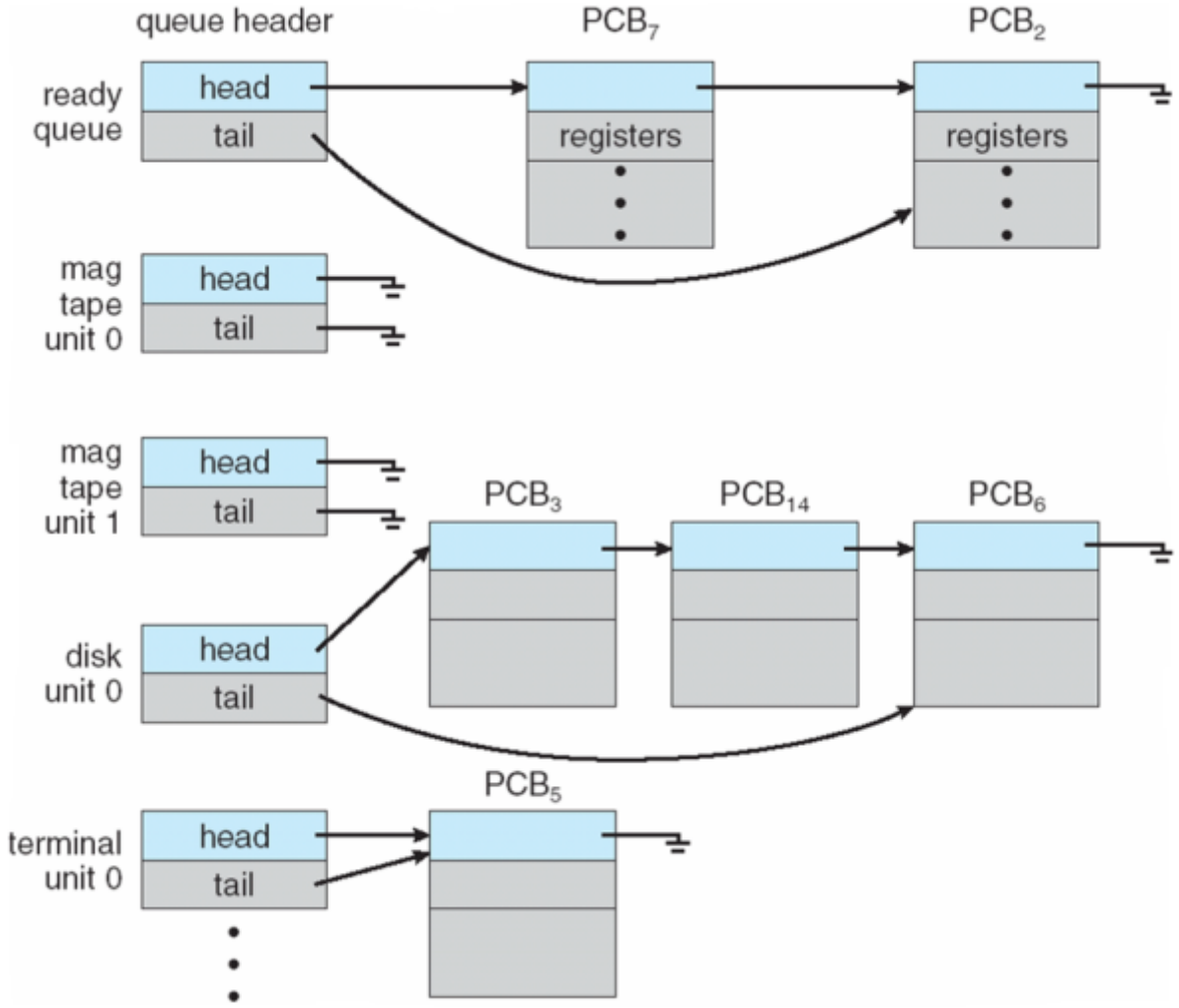
A medida que los procesos entran en el sistema, se colocan en una cola de trabajos que contiene

todos los procesos del sistema. Los procesos que residen en la memoria principal y están preparados y en espera de ejecutarse se mantienen en una lista denominada cola de procesos preparados. Generalmente, esta cola se almacena en forma de lista enlazada. La cabecera de la cola de procesos preparados contiene punteros al primer y último bloques de control de procesos (PCB) de la lista. Cada PCB incluye un campo de puntero que apunta al siguiente PCB de la cola de procesos preparados.

El sistema también incluye otras colas. Cuando se asigna la CPU a un proceso, éste se ejecuta durante un rato y finalmente termina, es interrumpido o espera a que se produzca un determinado suceso, como la terminación de una solicitud de E/S. Suponga que el proceso hace una solicitud de E/S a un dispositivo compartido, como por ejemplo un disco. Dado que hay muchos procesos en el sistema, el disco puede estar ocupado con la solicitud de E/S de algún otro proceso. Por tanto, nuestro proceso puede tener que esperar para poder acceder al disco. La lista de procesos en espera de un determinado dispositivo de E/S se denomina cola del dispositivo. Cada dispositivo tiene su propia cola.

Una representación que habitualmente se emplea para explicar la planificación de procesos es el diagrama de colas, como el mostrado en la Figura, donde cada rectángulo representa una cola. Hay dos tipos de colas: la cola de procesos preparados y un conjunto de colas de dispositivo. Los círculos representan los recursos que dan servicio a las colas y las flechas Indican el flujo de procesos en el sistema.






Cada proceso nuevo se coloca inicialmente en la cola de procesos preparados, donde espera hasta que es seleccionado para ejecución, es decir, hasta que es despachado. Una vez que se asigna la CPU al proceso y éste comienza a ejecutarse, se puede producir uno de los sucesos siguientes:

- El proceso podría ejecutar una solicitud de E/S y ser colocado, como consecuencia, en una cola de E/S.
- El proceso podría crear un nuevo subproceso y esperar a que éste termine.
- El proceso podría ser desalojado de la CPU como resultado de una interrupción y puesto de nuevo en la cola de procesos preparados.

En los dos primeros casos, el proceso terminará, antes o después, por cambiar del estado de espera al estado preparado y será colocado de nuevo en la cola de procesos preparados. Los procesos siguen este ciclo hasta que termina su ejecución, momento en el que se elimina el proceso de todas las colas y se desasignan su PCB y sus recursos.

## ¿Qué es concurrencia?

Dos o más procesos decimos que son concurrentes o que se ejecutan concurrentemente, cuando son procesados al mismo tiempo (no a nivel procesador), es decir, van avanzando los dos en su ejecución ( el “Instruction Pointer” va avanzando para los dos procesos), pero no son ejecutados en el procesador al mismo tiempo.

En sistemas multiprocesador, esta ejecución simultánea (paralela) podría conseguirse

completamente, puesto que podremos asignarle, por ejemplo, un proceso A al procesador A y un proceso B al procesador B y cada procesador realizaran la ejecución de su proceso.

Cuando tenemos un solo procesador se producirá un intercalado de las instrucciones de ambos procesos, de tal forma que tendremos la sensación de que se ejecutan juntos en el sistema (concurrencia, ejecución simultánea de más de un proceso).

Ahora bien, está claro que en esto tenemos que tener en cuenta que mientras un proceso está escribiendo un valor en una variable determinada, puede darse el caso que otro proceso que es concurrente al primero vaya a leer o escribir en esa misma variable, entonces habrá que estudiar el caso en el que un proceso haga una operación sobre una variable (o recurso en general) y otro proceso concurrente a él realice otra operación de tal forma que no se realice correctamente.

From:
http://wiki.educabit.ar/ - **Wiki Sistemas**

Permanent link:
**http://wiki.educabit.ar/doku.php?id=so_procesos**

Last update: **2025/09/11 22:48**