

ADMINISTRACIÓN DE RECURSOS

Empecemos por conocer algunos conceptos comúnmente utilizados en esta materia:

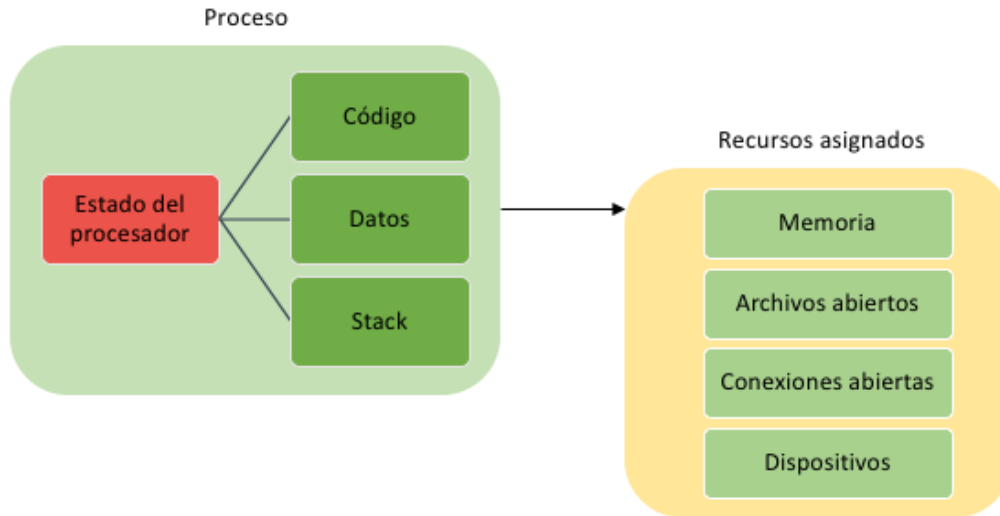
- **Procesos y Flujos.**

El **proceso** hace referencia a un programa en ejecución. A los procesos, dependiendo especialmente del sistema operativo utilizado, se les denomina **flujos** de control, tareas, threads o hilos, según el contexto. Cuando se ejecuta más de un proceso de forma concurrente en un sistema, cada proceso necesita que el propio sistema le suministre una serie de recursos. Para ello, el sistema operativo, gracias a la UCP (Unidad Central de Proceso), se encarga de asignar estos recursos en un orden adecuado y atendiendo ciertas prioridades. También realiza funciones de sincronización de todos los procesos, para que se ejecuten en el orden adecuado y según la prioridad decidida.

Cada vez que un programa se convierte en proceso, es decir, cada vez que se ejecuta un programa, además de ubicar en memoria las instrucciones que lo componen y sus datos asociados, a dicho proceso se le asocia una estructura de datos. Esta estructura, única para cada proceso, identifica el proceso respecto de los demás y sirve para controlar su correcta ejecución. A esto se le conoce como **bloque de control del proceso** o **BCP** y contendrá la siguiente información para cada proceso: estado actual del proceso, identificador del proceso, prioridad del proceso, ubicación en memoria y recursos utilizados.

Características de un Proceso:

- Los servicios superiores del SO se estructuran en base de procesos.
- Permite modularizar y aislar fallas de programas durante su ejecución.
- Soporta concurrencia de actividades, lo que permite un mejor aprovechamiento de los recursos.
- Denominaremos procesos a los trabajos (jobs) en sistema de lotes, como a las tareas en sistema compartido.

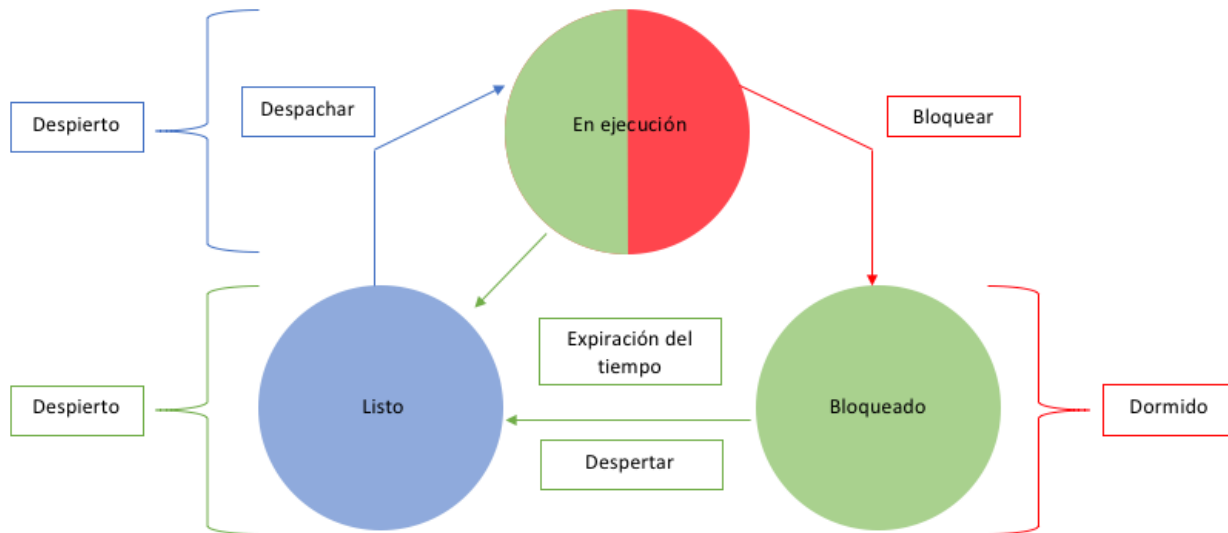


- **Hebras y estados de los procesos**

Una **hebra** es un punto de ejecución de un proceso. Un proceso tendrá siempre una hebra, en la que corre el propio programa, pero puede tener más hebras. Las hebras representan un método software para mejorar el rendimiento y eficacia de los sistemas operativos. Las hebras de un mismo proceso compartirán recursos, como memoria, archivos, recursos hardware, etcétera.

Básicamente los **estados** posibles de un proceso son los siguientes:

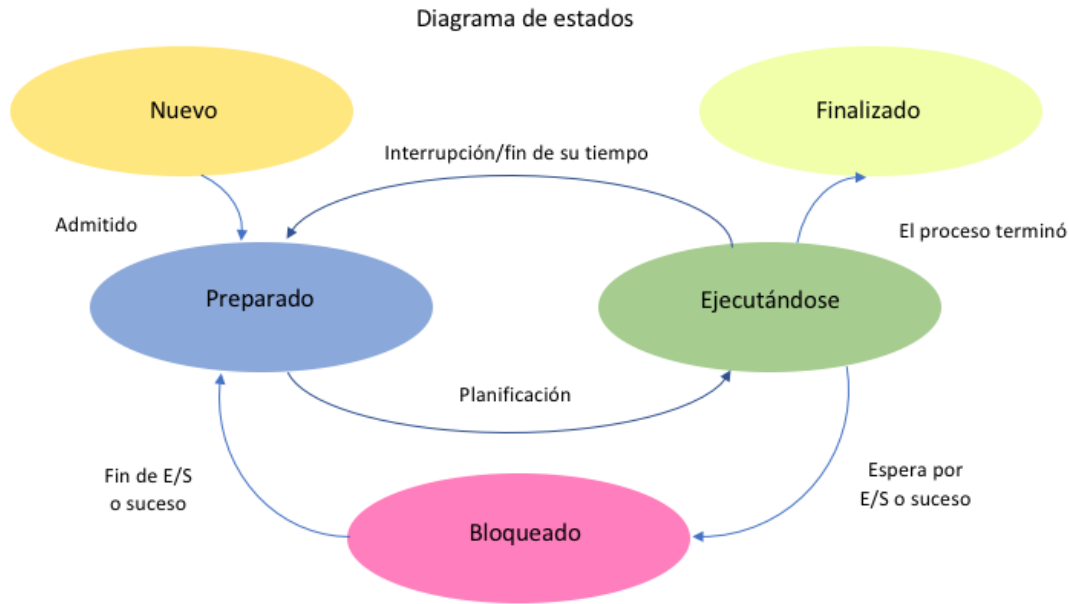
- En ejecución. El procesador está ejecutando instrucciones del programa que lo compone y tiene concedido el tiempo de uso de la UCP en un instante concreto.
- Preparado, en espera o activo. Un proceso está preparado para ser ejecutado; es decir, está esperando turno para poder utilizar su intervalo de tiempo y poner en funcionamiento sus instrucciones accediendo a los recursos del sistema.
- Bloqueado. El proceso está retenido; es decir, está bloqueado debido a causas múltiples. Una de estas causas puede ser que dos procesos utilicen el mismo fichero de datos. Otra puede ser que dos procesos necesiten utilizar la misma unidad de CD-ROM para cargar determinados datos, etcétera.



En general, todos los procesos dentro de cualquier sistema operativo tienen unas características que los identifican. En primer lugar, indicaremos que cada programa en ejecución, es decir, cada proceso, tiene un identificador que lo discrimina de los demás. Cada proceso tiene un número asignado por el sistema operativo que sirve precisamente para identificarlo, lanzarlo a ejecución, detenerlo, cancelarlo, reanudarlo, etcétera. Este identificador de proceso se nombra con la abreviatura PID. En cada sistema operativo, los procesos los lanzan normalmente otros procesos. Es decir, que cada proceso que se lanza a ejecución depende, en la mayoría de los casos, de otro proceso denominado **proceso padre**. Así, al nuevo proceso lanzado se le denomina proceso hijo.

- **Transición de los procesos**

Una vez que un programa se ha lanzado y se ha convertido en proceso, puede atravesar varias fases o estados hasta que finaliza o termina. Cuando un proceso se lanza, nunca se ejecuta directamente, sino que se coloca en la cola de procesos en un estado denominado preparado. Cuando la UCP le asigna su tiempo, el proceso pasa de preparado a ejecución. Estos dos estados se alternarán en caso de que se esté ejecutando más de un proceso en el sistema. Los cambios de estado en los que se puede encontrar un proceso se denominan transiciones. En la siguiente ilustración se recogen las transiciones o cambios de estado que pueden experimentar los procesos:



- Transición de Ejecución ha Bloqueado: ocurre cuando el programa que está en ejecución necesita algún elemento, señal o dato para continuar ejecutándose.
- Transición de ejecución a Listo: ocurre cuando un programa o proceso ha utilizado el tiempo asignado por la UCP (procesador) para su ejecución y tiene que dejar paso al siguiente proceso.
- Transición de Listo a Ejecución: ocurre cuando el proceso que está preparado pasa al proceso de ejecución, es decir, cuando al proceso le llega una nueva disposición de tiempo de la UCP para poder ejecutarse.
- Transición de Bloqueado a Preparado: ocurre cuando el proceso pasa de estar bloqueado a estar preparado, es decir, cuando el proceso recibe una orden o señal que estaba esperando para pasar al estado de preparado y, posteriormente, tras la transición, a estado de ejecución.

Los diferentes estados tienen una relación directa con lo que vamos a denominar **prioridades**, que son aquellas que el administrador del sistema o el propio sistema, asignan a cada proceso. De ello dependerá que un proceso se ejecute en más o menos tiempo.

Se pueden establecer prioridades en función de la necesidad de ejecución de algunos programas. Los programas que más se ejecutan, es decir, los más necesarios, tendrán prioridad de ejecución sobre aquellos que se ejecutan muy de cuando en cuando. Es ahora cuando hemos de hablar de la planificación. Con esta técnica conseguimos indicar al ordenador los procesos que deben ejecutarse y los estados que estos deben adoptar.

Gracias a los algoritmos de planificación podemos decidir qué proceso ha de ejecutarse en cada momento y por qué. Algunas características de estos algoritmos son la imparcialidad, la equidad, la eficiencia, el tiempo de respuesta y el rendimiento.

Para ejecutar un proceso, introducimos en el indicador de comandos (shell en UNIX, command.com en Windows) el nombre de un fichero ejecutable o hacemos doble clic con el ratón sobre el icono que representa un programa ejecutable. De esta forma, el sistema operativo prepara el programa a través del cargador para lanzarlo a ejecución.

Una vez cargado el proceso, el sistema operativo asigna a través del planificador la prioridad del nuevo proceso respecto de los que hay en ejecución. De esta forma, cada proceso atraviesa varias fases. En un momento dado, el proceso se estará ejecutando; posteriormente estará en espera, mientras la UCP ejecuta otro; otros procesos estarán preparados para ser lanzados; otros podrán estar bloqueados, etcétera. En estos cambios de proceso, el sistema operativo tiene que saber qué ficheros están abiertos en cada proceso y qué periféricos se están utilizando. Cuando se están ejecutando varias tareas (procesos) a la vez, es necesario compartir el tiempo de trabajo de la UCP. El tiempo compartido consiste en dividir el tiempo de ejecución del procesador en minúsculos intervalos de tiempo (quantum) e ir asignando cada uno de esos intervalos de ejecución a cada proceso que está en ejecución.

Referencia:

Pérez, Ángela. (2013). *Administración de Recursos*.

Recuperado de: <https://chsos20132912059.wordpress.com/2013/10/06/administracion-de-recursos/>