



## Hilos:

En el ambiente de los sistemas operativos se suele dividir el concepto de proceso en dos partes asociando cada una de ellas a una funcionalidad de los mismos. Por un lado denominamos **hilo (thread) o proceso ligero** a la unidad que se activa o que se ejecuta y por otro lado denominamos **proceso o tarea** a la unidad de propiedad de recursos. Esta última incluye la imagen del proceso (programa, datos, pila y atributos definidos en el bloque de control de proceso) en memoria principal, canales de E/S, dispositivos de E/S y archivos.

Multihilo se refiere a la capacidad de un SO a dar soporte a múltiples hilos de ejecución en un solo proceso. El enfoque tradicional se conoce como monohilo. Algunos ejemplos de SO monohilo son MS-DOS y algunas versiones de UNIX.

En un entorno multihilo se define la unidad de asignación de recursos y una de protección. Se asocian con los procesos los siguientes:

- Un espacio de direcciones virtuales, que contiene la imagen del proceso.
- Acceso protegido a los procesadores, otros procesos (para la comunicación entre procesos), archivos y recursos de E/S (dispositivos y canales)

En un proceso pueden haber uno o más hilos, cada uno con lo siguiente:

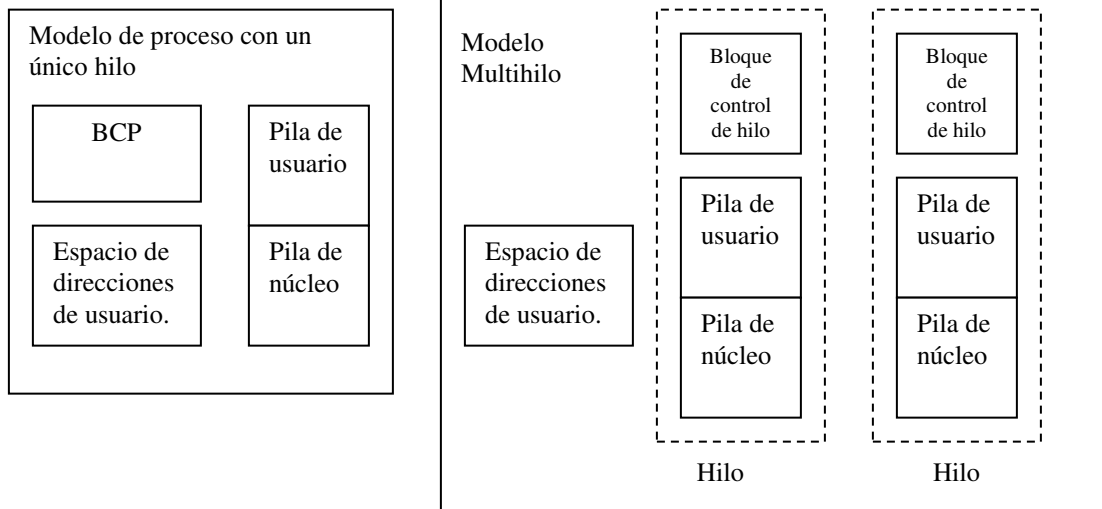
- El estado de ejecución del hilo (Ejecución, Listo, etc.)
- El contexto de hilo que se salva cuando no está ejecutando; una forma de contemplar al hilo es con un contador de programa independiente operando dentro de un proceso.
- Una pila de ejecución
- Por cada hilo, almacenamiento estático para las variables locales
- Acceso a la memoria y a los recursos de su proceso, que se comparten con todos los otros hilos del proceso.

En un entorno multihilo sigue habiendo un único bloque de control de proceso y un espacio de direcciones de usuario asociado al proceso, pero ahora hay varias pilas separadas para cada hilo, así como un bloque de control para cada hilo que contiene los valores de los registros, la prioridad y otra información relativa al estado del hilo.

Todos los hilos de un proceso comparten el estado y los recursos de ese proceso, residen en el mismo espacio de direcciones y tienen acceso a los mismos datos. Cuando un hilo cambia determinados datos en memoria, otros hilos ven los resultados cuando acceden a estos datos. Si un hilo abre un archivo con permiso de lectura, los demás hilos pueden leer también este archivo.

Los mayores beneficios provienen del rendimiento:

- 1- Lleva mucho menos tiempo crear un nuevo hilo en un procesos existente que crear un nuevo proceso totalmente nuevo.
- 2- Lleva menos tiempo finalizar un hilo que un proceso.
- 3- Lleva menos tiempo cambiar entre dos hilos dentro del mismo proceso.
- 4- Los hilos mejoran la eficiencia de la comunicación entre diferentes programas que están ejecutando. Los hilos dentro de un mismo proceso comparten memoria y archivos se pueden comunicar entre ellos sin necesidad de invocar al núcleo del SO.



Ejemplos de uso de los hilos en un sistema de multitarea:

- **Trabajo en primer plano y segundo plano:** Esto se produce en el sentido de la interacción directa con el usuario, no en el de sesiones interactivas y de fondo. Por ejemplo, en un programa de hoja de cálculo, un hilo puede estar visualizando los menús y leyendo la entrada del usuario mientras que otro hilo ejecuta las órdenes y actualiza la hoja de cálculo. Esta medida suele aumentar la velocidad que se percibe de la aplicación, permitiendo que el programa pida la orden siguiente antes de terminar la anterior.
- **Procesamiento asíncrono:** Los elementos asíncronos del programa se pueden implementar como hilos. Por ejemplo, para protegerse de un corte de alimentación, se puede diseñar un procesador de textos que escriba su buffer de la RAM al disco una vez por minuto. Se puede crear un hilo cuya única tarea sea hacer estas copias de respaldo periódicas y que se planifique directamente con el sistema operativo; no hay necesidad de ningún código superfluo en el programa principal que haga la comprobación de tiempo o que coordine la entrada y la salida.



- **Velocidad de la ejecución:** Un proceso con hilos múltiples puede computar un lote de datos mientras lee el lote siguiente de un dispositivo. En un sistema con multiproceso, varios hilos de un mismo proceso podrán ejecutar realmente a la vez.
- **Estructura modular u organización de los programas:** Los programas que suponen una variedad de actividades o varios orígenes y destinos de entrada y salida pueden hacerse más fáciles de diseñar e implementar mediante hilos.

La planificación y la expedición se llevan a cabo con los hilos; por tanto, la mayor parte de la información de estado relacionada con la ejecución se mantiene en estructuras de datos al nivel de los hilos. Sin embargo, hay varias acciones que afectan a todos los hilos de una tarea y que el sistema operativo debe gestionar al nivel de las tareas. La suspensión implica la descarga del espacio de direcciones fuera de la memoria principal. Puesto que todos los hilos de una tarea comparten el mismo espacio de direcciones, todos deben entrar en el estado Suspendido al mismo tiempo. De manera similar, la terminación de una tarea supone terminar con todos los hilos dentro de dicha tarea.

Funcionalidades de los hilos:

Los hilos tienen estados de ejecución y se pueden sincronizar entre ellos.

Estados de los hilos:

Los estados de los hilos son: Ejecución, Listo y Bloqueado. El estado suspendido se aplica a los procesos y no a los hilos.

Las operaciones básicas relacionadas a los hilos están asociadas al cambio de estado del hilo:

- **Creación:** cuando se crea un nuevo proceso, también se crea un hilo de dicho proceso. Un hilo posteriormente puede crear otros hilos dentro del proceso proporcionando un puntero a las instrucciones y los argumentos para el nuevo hilo. Se le proporciona su propio registro de contexto y espacio de pila y se lo coloca en la cola de Listos.
- **Bloqueo:** cuando un hilo necesita esperar por un evento se bloquea, almacenando los registros de usuario, contador de programa y puntero de pila. El procesador puede pasar a ejecutar otro hilo en estado listo dentro del mismo proceso o en otro diferente.
- **Desbloqueo:** Cuando sucede el evento por el que el hilo está bloqueado, el hilo se pasa a cola de Listos.
- **Finalización:** Cuando se completa un hilo, se liberan su registro de contexto y pilas.



En un uniprosesador, la multiprogramación permite el intercalado de múltiples hilos con múltiples procesos.

Sincronización de hilos:

Todos los hilos de un proceso comparten el mismo espacio de direcciones y otros recursos. Cualquier alteración de un recurso por cualquiera de los hilos, afecta al entorno del resto de los hilos del mismo proceso. Por lo tanto, es necesario sincronizar las actividades de los hilos para que no interfieran entre ellos o corrompan estructuras de datos.