**Hilos (threads):**

Los hilos son un concepto relativamente nuevo de los SO. En este contexto, un proceso recibe el nombre de proceso pesado, mientras que un hilo recibe el nombre de proceso ligero.

El término hilo se refiere sintáctica y semánticamente a hilos de ejecución.

El término multihilo hace referencia a la capacidad de un SO para mantener varios hilos de ejecución dentro del mismo proceso.

En un SO con procesos mono hilo (un solo hilo de ejecución por proceso), en el que no existe el concepto de hilo, la representación de un proceso incluye su PCB, un espacio de direcciones del proceso, una pila de proceso y una pila núcleo.



En un SO con procesos multihilo, sólo hay un PCB y un espacio de direcciones asociados al proceso, sin embargo, ahora hay pilas separadas para cada hiloy bloques de control para cada hilo.



**Estructura de los hilos:**

Un hilo (proceso ligero) es una unidad básica de utilización de la CPU, y consiste en un contador de programa, un juego de registros y un espacio de pila.

Los hilos dentro de una misma aplicación comparten:

* La sección de código.
* La sección de datos.
* Los recursos del SO (archivos abiertos y señales).

Un proceso tradicional o pesado es igual a una tarea con un solo hilo.

Los hilos permiten la ejecución concurrente de varias secuencias de instrucciones asociadas a diferentes funciones dentro de un mismo proceso, compartiendo un mismo espacio de direcciones y las mismas estructuras de datos del núcleo.

* Recursos compartidos entre los hilos:
* Código(instrucciones).
* Variables globales.
* Ficheros y dispositivos abiertos.
* Recursos no compartidos entre los hilos:
* Contador del programa (cada hilo puede ejecutar una sección distinta de código).
* Registros de CPU.
* Pila para las variables locales de los procedimientos a las que se invoca después de crear un hilo.
* Estado: distintos hilos pueden estar en ejecución, listos o bloqueados esperando un evento.

**Estado Hilos:**

Los principales estados de un hilo son: *ejecución, preparado y bloqueado* y hay cuatro operaciones básicas relacionadas con el cambio de estado de los hilos:

Creación: En general, cuando se crea un nuevo proceso se crea también un hilo para ese proceso. Posteriormente, ese hilo puede crear nuevos hilos dándoles un puntero de instrucción y algunos argumentos. Ese hilo se colocará en la cola de preparados.

Bloqueo: Cuando un hilo debe esperar por un suceso, se le bloquea guardando sus registros. Así el procesador pasará a ejecutar otro hilo preparado.

Desbloqueo: Cuando se produce el suceso por el que un hilo se bloqueó pasa a la cola de listos.

Terminación: Cuando un hilo finaliza, se liberan su contexto y sus pilas.

Nota: Un punto importante es la posibilidad de que el bloqueo de un hilo lleve al bloqueo de todo el proceso. Es decir, que el bloqueo de un hilo lleve al bloqueo de todos los hilos que lo componen, aun cuando el proceso está preparado.

En un sistema monoprocesador, la multiprogramación permite intercalar la ejecución de múltiples hilos dentro del mismo proceso:



Los hilos operan, en muchos sentidos, igual que los procesos:

* Pueden estar en uno o varios estados: listo, bloqueado, en ejecución o terminado.
* También comparten la CPU.
* Sólo hay un hilo activo (en ejecución) en un instante dado.
* Un hilo dentro de un proceso se ejecuta secuencialmente.
* Cada hilo tiene su propia pila y contador de programa.
* Pueden crear sus propios hilos hijos.

**Diferencia con los procesos:**

Los hilos no son independientes entre sí. Como todos los hilos pueden acceder a todas las direcciones de la tarea, un hilo puede leer la pila de cualquier otro hilo o escribir sobre ella. Aunque pueda parecer lo contrario la protección no es necesaria ya que el diseño de una tarea con múltiples hilos tiene que ser un usuario único.

**Ventajas de los hilos sobre los procesos:**

* Se tarda mucho menos tiempo en crear un nuevo hilo en un proceso existente que en crear un nuevo proceso.
* Se tarda mucho menos tiempo en terminar un hiloque un proceso.
* Se tarda mucho menos tiempo en conmutar entre hilosde un mismo proceso que entre procesos.
* Los hilos hacen más rápida la comunicación entre procesos, ya que, al compartir memoria y recursos, se pueden comunicar entre sí sin invocar el núcleo del SO.

**Ejemplos de uso de los hilos:**

* Trabajo interactivo y en segundo plano:En un programa de hoja de cálculo, un hilo podría estar leyendo la entrada del usuario y otro podría estar ejecutando las órdenes y actualizando la información.
* Procesamiento asíncrono: Se podría implementar, con el fin de protegerse de cortes de energía, un hilo que se encargará de salvaguardar el buffer de un procesador de textos una vez por minuto.
* Estructuración modular de los programas: Los programas que realizan una variedad de actividades se pueden diseñar e implementar mediante hilos.