

Hilos

Andrés Felipe Rodríguez Ferrer

Informatica II

1020496316

Julio 2020

Un hilo, se les puede llamar como procesos ligeros o subprocesos, estos constituyen la ejecución de un proceso. Estos procesos pueden estar formados por un monohilo (un solo hilo) o multihilo (varios hilos). Todos los hilos que pertenecen a un proceso tienen un entorno igual de ejecución (variables, direcciones, memoria, ficheros abiertos, etc.), pero cada hilo, si tiene sus propios registros en la CPU, pila y variables locales, que permitirán que cuando regrese al procesador, la ejecución se pueda proseguir en la línea que quedo. Desde la perspectiva de la programación, son como funciones que se pueden lanzar en paralelo con otras. Sin embargo, los hilos son mencionados más en el ámbito de los procesadores, donde la teoría que se implementó en ellos, se logró llevar a los microprocesadores y que también utilizaran los hilos.[3]

La historia de los hilos se remonta hasta 1965, con el sistema Berkeley Time-sharing, donde Dijkstra, los llamo procesos y no hilos. En el mismo año IBM, con su lenguaje de programación PL/I, desarrollo una característica que bifurcaba a una función creando un hilo, pero no se sabe si algún compilador de la empresa implemento esta característica, en las versiones posteriores, se eliminó este llamado, pero se estudió con detalle antes de quitarlo. En la década de los 70, Unix empezaron a reemplazar los procesos que compartían memoria por subprocesos que compartían el espacio de direcciones de un único proceso de Unix. En un inicio fueron llamados “livianos”, en contraste de los procesos “pesados”. Esta diferencia se produjo a finales de los 70 e inicios de los 80, donde aparecieron los “microkernels” [4].

Los hilos se pueden diferenciar en dos niveles:

1. Hilos a nivel de usuario: estos están implementados en algunas librerías. Estos hilos se administran sin necesidad del sistema operativo, este solo reconoce un hilo de ejecución. Estos hilos tienen como ventaja que su cambio de contexto es más simplificado que el cambio de contexto de los hilos del kernel.

Además, se pueden llevar acabo sin la necesidad del sistema operativo. otra de las ventajas que este tipo de hilos tienen es en poder plantear diferentes planificaciones del sistema operativo.[2]

2. Hilos a nivel del kernel: este tipo son directamente creados por el sistema operativo, el cual es quien los planifica y gestión. Tiene como gran ventaja el aprovechar mejor las arquitecturas de los microprocesadores, esto proporciona un mejor tiempo de respuesta, ya que, si un hilo se bloquea, los otros siguen trabajando, su ejecución no depende del otro. [2]

Muchos lenguajes de programación soportan algunos hilos, pero en otros es necesario implementarlos como es el caso de los lenguajes C y C++, en estos lenguajes es necesario crearlos por medio de bibliotecas especiales que dependen del sistema operativo en el que estén siendo utilizados. Como se ve muchas veces, por no decir en el mayor de los casos, el sistema operativo afecta el desarrollo y las implementaciones de los hilos, lo que quiere decir que la arquitectura del sistema operativo afectara el manejo y creación de hilos a la hora querer hacer un programa.[1]

References

- [1] C0x66. *Hilos*. 2017.
- [2] Universidad de la Republica-Uruguay. *Hilos*. Universidad de la Republica-Uruguay.
- [3] J. Villar F. Huércano. *Desarrollo de componentes software para servicios de comunicaciones. IFCT0609*. IC Editorial, 2014.
- [4] B. OSullivan. *The histoy of threads*. 2005.