

Hilos

Victor Manuel Arbeláez Ramírez

Facultad de ingeniería

Universidad de Antioquia

Un hilo, es un medio que administra el flujo de control de datos de un programa, separándolo en tareas que son realizadas por el microprocesador de una manera más eficiente con la ayuda de sus núcleos, gracias a la tareas o procesos generados por el programa. En otras palabras, se divide en "trozos", para optimizar los tiempos de espera de los subprocesos o threads.

Los hilos, threads o subprocesos no forman parte física del procesador, algo que habitualmente es malinterpretado, los núcleos existen a nivel de hardware y los hilos a nivel de software, ayudando a los núcleos del microprocesador a ser mucho más eficaces llevando a cabo múltiples tareas simultáneamente.

En síntesis, las principales ventajas de la programación multihilo, se dividen en la capacidad de respuesta, la compartición de recursos, la economía, y el correcto uso de la arquitectura de los procesadores o microprocesadores.

Tomando una visión al pasado, inicialmente el significado de un hilo, como un flujo secuencial de control, se da en 1965 con el sistema operativo Berkeley Timesharing, aunque en ese momento fueron llamados procesos y no hilos e interactuaban a través de variables compartidas.

Se piensa que el lenguaje PL / 1 fue uno de los más importantes precursores en el ámbito de lo hilos, aunque no se tiene la certeza de que algún compilador de IBM pudiese hacer uso de él, sin embargo, existen registros de que se hicieron constantes pruebas en el sistema operativo MULTICS.

Después vino Unix, en la década de 1970. La noción de Unix de un "proceso", era en un sentido "pesado" ya que no podían compartir memoria, interactuaban a través de tuberías, señales, etc, convirtiéndose así en un hilo de control secuencial más un espacio de direcciones virtuales y posteriormente, se condujo a la "invención" de subprocesos, los cuales se conocieron como "livianos", en contraste con los procesos "pesados" de Unix, dándose esta distinción a partir de los años 80, cuando aparecieron los primeros "microkernels".

En los modelos multihilo, se encuentran las implementaciones más comunes, en donde el soporte para los hilos se da en dos principales categorías, en el nivel de usuario (ULT) o nivel del kernel (KLT).

Para establecer relaciones entre los hilos a nivel de usuario y de kernel, se encuentran los siguientes modelos:

- El modelo muchos a uno, donde se asignan múltiples hilos del nivel de usuario a un hilo del nivel de kernel.
- El modelo uno a uno, donde se asigna cada hilo de usuario a un hilo de nivel de kernel, permitiendo la ejecución múltiple de hilos en paralelo en distintos procesadores.
- El modelo muchos a muchos donde se transmiten muchos hilos de usuario sobre un número menor o igual de hilos de kernel, siendo éste el modelo más eficiente.

También se puede dar la combinación de ULT y KLT en algunos sistemas operativos como lo es Solaris. En este caso, el programador, puede ajustar los parámetros para buscar el mejor resultado global en la creación de hilos, así como la mayor parte de la planificación y la sincronización es llevada a cabo en el espacio de usuario donde los distintos ULT de una aplicación se relacionan con varios KLT.

En las aplicaciones de hilos a nivel del kernel (KLT) pura, todo el trabajo para la gestión de hilos es realizada por el Kernel excluyendo toda participación del nivel de usuario. En el área de la aplicación solo hay una interfaz de programas de aplicación para la gestión de hilos en el núcleo. Windows 2000, Linux y OS/2 utilizan este método, aunque Linux se diferencia en que no hace distinciones entre los procesos e hilos.

Existe también el módulo KVM del tipo KLT, donde con el lenguaje de programación Java se proporciona una interfaz para que el programador acceda a las funciones nativas del kernel y lograr gestionar los hilos que puedan estar ejecutándose; siendo este modelo, independiente del hardware, portable e independiente del lenguaje Java.

En una aplicación de Hilos a nivel del usuario (ULT) pura, todo el trabajo de gestión de hilos es realizado por la aplicación, mientras el kernel no es consciente de la existencia de estos hilos. Esta aplicación es programable mediante la biblioteca de hilos, con sus operaciones llevadas a cabo en el espacio del usuario de un mismo proceso.

Esta implementación toma ventaja en la medida que no necesita los privilegios del modo kernel, ya que las estructuras de datos están almacenadas en el espacio de direcciones de un mismo proceso evitando así una sobrecarga u overhead. También se pueden ejecutar en cualquier sistema operativo, teniendo en cuenta que la biblioteca de hilos es un conjunto compartido.

Por el contrario, hay una gran desventaja, en las estrategias ULT puras, una aplicación multihilo no puede aprovechar las opciones del hardware de los procesadores y se asigna un solo proceso a éstos, excluyendo la participación eficiente de los núcleos.

Referencias

- Castillo, J. A. (2019). *Hilos de un procesador*. Recuperado el 16 de julio de 2020, desde <https://www.profesionalreview.com/2019/04/03/que-son-los-hilos-de-un-procesador/#:~:text=Qu%C3%A9%20son%20los%20hilos%20de%20procesamiento%20o%20threads,-Los%20hilos%2C%20threads&text=Podemos%20definir%20un%20hilo%20de,de%20una%20forma%20m%C3%A1s%20eficiente.>
- García, E. R. (2017). *Núcleos e hilos en un procesador: qué son y en qué se diferencian*. Recuperado el 16 de julio de 2020, desde https://www.elespanol.com/omicrono/tecnologia/20170707/nucleos-hilos-procesador-diferencian/229478224_0.html
- O'Sullivan, B. (2005). *The history of threads*. Recuperado el 16 de julio de 2020, desde <http://www.serpentine.com/blog/threads-faq/the-history-of-threads/>

Patterson, D. A. & Hennessy, J. L. (2018). *Estructura y diseño de computadores*.

Recuperado el 16 de julio de 2020, desde <https://books.google.com.co/books?id=YDbeDwAAQBAJ&pg=PA646&dq=hilos+procesador&hl=es-419&sa=X&ved=2ahUKEwi1jYq37srqAhXEc98KHf4FA4QQ6AEwAnoECAQQAg#v=onepage&q=hilos%20&f=false>

Rodríguez, L. D. (2007). *El Gran Libro del PC Interno*. Recuperado el 16 de julio de 2020, desde <https://books.google.com.co/books?id=g7SoQDLyXrYC&pg=PA44&dq=hilos+microprocesadores&hl=es-419&sa=X&ved=2ahUKEwih5-mh7crqAhXGnuAKHXFtCpQQ6AEwAXoECAIQAg#v=onepage&q=hilos%20microprocesadores&f=false>

Turmero, P. (s.f.). *Hilos de procesamiento*. Recuperado el 16 de julio de 2020, desde <https://aprendiendoarduino.wordpress.com/2016/11/13/interrupciones/>