Instituto Tecnológico de Colima

División de Estudios de Posgrado e Investigación

Maestría en Sistemas Computacionales

Proyecto:

**HILOS**

Materia de Tecnologías de Programación

Presenta:

Gabriel López Sánchez

Profesor:

Francisco Cervantes Zambrano

**Villa de Álvarez, Colima. 16 de Mayo de 2018.**

**Hilos**

Los hilos de ejecución son segmentos de código de un programa que se ejecutan secuencialmente de modo independiente de las otras partes del programa. La principal utilización de thread en un programa es para el aislamiento y sincronización de tareas.

Un hilo (thread) es un proceso en ejecucion dentro de un programa.

* La finalizacion depende del hilo
* Los hilos implementan prioridad y mecanismos de sincronizacion.
* Cualquier clase se puede hacer hilo

**Demonios**

Los hilos de ejecución demonio también se llaman servicios, porque se ejecutan, normalmente, com prioridad baja y proporcionan un servicio básico a un programa o programas cuando la actividad de la maquina es reducida. Los hilos demonio son utiles cuando un hilo debe ejecutarse en segundo plano durante largos periodos de tiempo, un ejemplo de hilo demonio que este ejecutándose continuamente es él recolector de basura (garbage collector) Este hilo, proporcionado por la Maquina virtual Java comprueba las variables de los programas a las que no se accede nunca y libera estos recursos, devolviéndolos al sistema.

**Proceso**

Llamamos proceso a la ejecución de un programa secuencial.

Un hilo o proceso en Java comienza con una instancia de la clase java.lang.Thread, si analizamos la estructura de dicha clase podremos encontrar bastantes métodos que nos ayudan a controlar el comportamiento de los hilos, desde crear un hilo, iniciarlo, pausar su ejecución, etc.

**Interbloqueo**

(deadlock): suceden cuando dos o más subprocesos en un programa no tienen acceso a objetos y están a la espera de ellos, los cuales nunca estarán disponibles, esto es, varios procesos compiten por los mismos recursos, pero ninguno los consigue.

**Dispatcher**

Java procesa todos los eventos de ratón y teclado en un hilo aparte conocido como hilo de despacho de eventos (EDT Event dispatch Thread). Este hilo es también el encargado de repintar las ventanas cuando las movemos, redimensionamos o pasamos otras ventanas por encima.

El problema de este mecanismo es que si ante un evento de ratón o teclado realizamos una tarea que tarde mucho tiempo, durante todo ese tiempo el EDT está haciendo nuestra tarea y, por tanto, no está repintando las ventanas ni atendiendo a otros eventos de teclado o ratón.

El resultado es que mientras realizamos esa tarea, nuestra ventana no responde a eventos de teclado o ratón y no se repinta si pasamos otra ventana por encima. Este efecto dura hasta que terminemos de realizar nuestra tarea. Todo lo que hemos intentado hacer con la ventana con el ratón o teclado queda "encolado", de forma que cuando termina nuestra tarea, todo se repinta de golpe.

**Bibliografia**

Hilos en java 2: <http://questgarden.com/97/28/1/100226080235/process.htm>

Hilos Demonio : <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwjOza_Mss3bAhVIRK0KHfq6D8gQFggnMAA&url=http%3A%2F%2Fpro-concurrente.galeon.com%2Finvc%2FHILOS-DEMONIO.doc&usg=AOvVaw1gLy_rKwhE145PDv1S6UoG>

Hilos: <http://www.jtech.ua.es/dadm/restringido/java/sesion05-apuntes.html>

El EDT: <http://chuwiki.chuidiang.org/index.php?title=El_EDT_(Event_Dispatch_Thread)_de_Java>