

Escuela Colombiana de Ingeniería

Arquitecturas de Software

programación concurrente, condiciones de carrera y sincronización de hilos.

Integrantes: Stefany Morón Castro - Juan Pablo Arévalo Merchán

Parte I – Antes de terminar la clase.

Control de hilos con wait/notify. Productor/consumidor.

1. Revise el funcionamiento del programa y ejecútalo. Mientras esto ocurre, ejecute jVisualVM y revise el consumo de CPU del proceso correspondiente.
 - A qué se debe este consumo?,

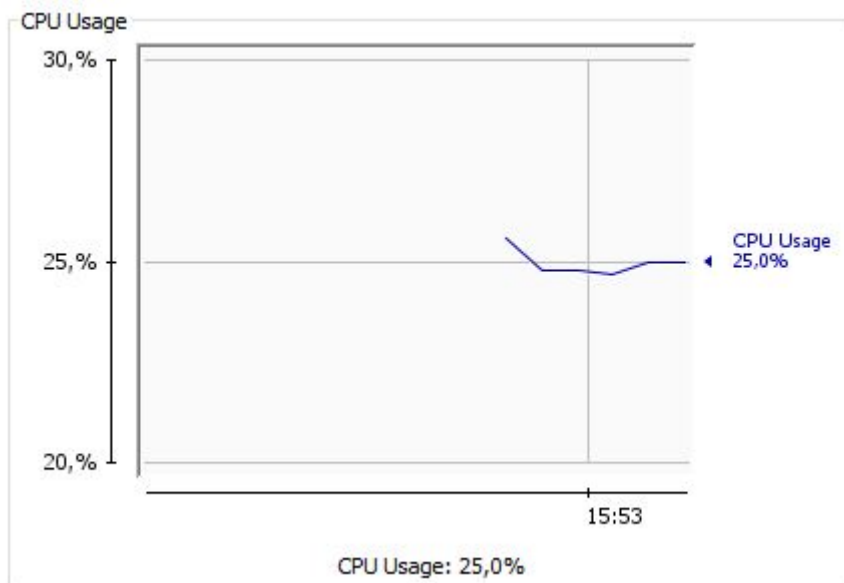
Lo primero que se debe tener en cuenta es que la clase `Producer`, se duerme durante un determinado tiempo lo que genera que se produzca más espacio de lo que se consume, por lo tanto la clase `consumer` está generando una espera activa constante y esto hace que haya mayor consumo de CPU.

- ¿Cuál es la clase responsable?

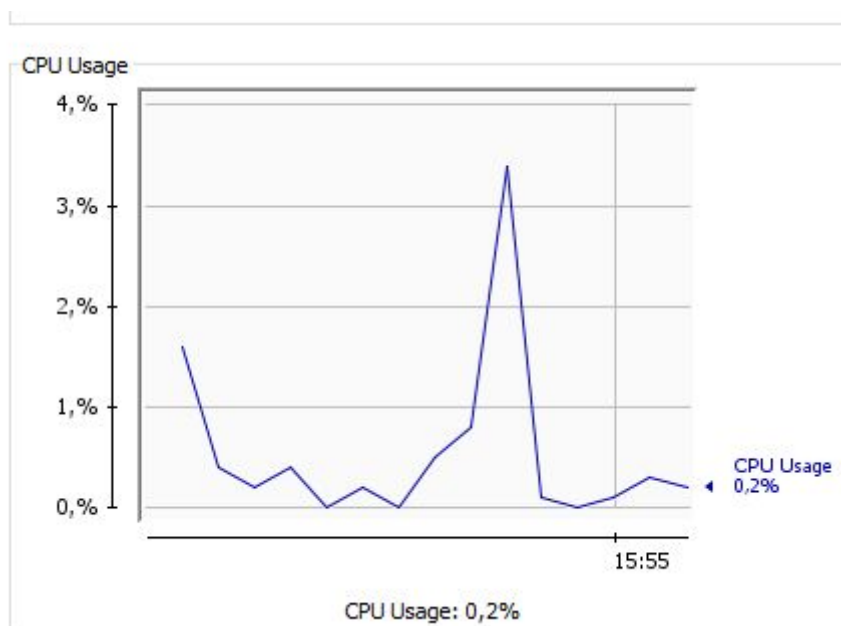
La clase responsable es `consumer` ya que está haciendo una consulta constante si hay productos para consumir.

2. Haga los ajustes necesarios para que la solución use más eficientemente la CPU, teniendo en cuenta que -por ahora- la producción es lenta y el consumo es rápido. Verifique con JVisualVM que el consumo de CPU se reduzca.

Al principio, el consumo de CPU era el siguiente:



consumía en promedio 25% de la CPU, con los cambio realizado, quitándole la espera activa que tenia, el consumo de CPU quedó así:



Un promedio mucho más bajo que el anterior.