

Hemos visto en el anterior post varias formas de usar PatternLayout a la hora de almacenar información sobre log para nuestra aplicación . Una de las opciones que hemos visto es usar %d que nos muestra la fecha . Sin embargo esta forma delega en el api del JDK y en la clase SimpleDateFormat que no tiene buen rendimiento . Es mejor usar alguno de los propios formateadores de Log4J para fechas . Cambiando nuestro código de %d a %d{ABSOLUTE} o %d{DATE} de esta forma ganaremos en rendimiento.

Otras dos opciones que son realmente lentas a la hora de grabar información al log son %M (nombre del método) y %F (nombre del fichero que generó el Log) evitemoslas dentro de lo posible .

Otra de las cosas que suele relentizar el funcionamiento del Log es el uso que hacemos por defecto de los appenders de consola

```
log4j.appender.miappender=org.apache.log4j.ConsoleAppender
```

Este tipo de appenders en un primer momento nos parecen muy útiles por ser muy sencillos por ejemplo yo los he usado en los distintos capítulos del libro . Sin embargo cuando llegamos a entornos de producción reales no son tan útiles ya que los administradores prefieren un appender a fichero FileAppender. Lo que mucha gente no conoce es que no solo es mas util sino que tambien es mucho mas rapido usar un FileAppender que un ConsoleAppender . Concretamente el FileAppender es el doble de rapido que el ConsoleAppender . Por lo tanto es interesante en producción desactivar los appender de Consola que tengamos.

El FileAppender como su nombre indica escribe en un fichero . Ahora bien la forma de escribir es continua es decir cada vez que llega un mensaje lo escribe en el fichero lo cual en un principio parece razonable pero implica también que estamos escribiendo continuamente . Esto se debe a que el FileAppender tiene configurada la siguiente opción por defecto.

```
log4j.appender.FILE.ImmediateFlush=true
```

Si cambiamos esta opción por false ganaremos rendimiento ya que el FileAppender usará un Buffer para cachear un conjunto de mensajes antes de escribirlos al disco . Simplemente activando esta opción el rendimiento del log puede mejorar un 50% o mas.