

MODIFICACIONES ENTREGA PRÁCTICA 5 PC

Francisco Javier Blázquez Martínez

Todos los cambios realizados tras la revisión de la práctica se enmarcan en [este commit](#). Explicamos estos a demás en este documento. Todos han sido en la parte del servidor y hemos conseguido aumentar el nivel de concurrencia permitida sin perder corrección ni dar lugar a posibles interbloqueos.

1. La clase **Server** deja de tener todos sus métodos estáticos y, en vez de ejecutarse únicamente mediante un método *main*, pasa a implementar la interfaz *Runnable*. De esta forma permitimos que puedan ejecutarse varios servidores en una misma *host-machine*.
2. En base al punto anterior es también necesario que la clase **Server** no emplee atributos estáticos (pues podríamos tener varios servidores distintos que no pueden compartir sus tablas/hashMaps de información). Se aplica esta modificación.
3. Punto más importante pues es responsable de aumentar el nivel de concurrencia que permite nuestro servidor. Con la implementación anterior con todos los métodos *synchronized*, la ejecución de cualquier método de la clase **Server** impedía temporalmente la ejecución de cualquier otro. Como algunos métodos podían coexistir hemos reemplazado los métodos *synchronized* por bloques *synchronized* explícitos para controlar el acceso concurrente a ciertos atributos sin restringir el acceso de forma innecesaria a otros.

No he creado clases monitores como tales para los atributos y he optado por monitores implícitos (especificando los bloques que se ejecutan con restricciones *synchronized* y sobre qué atributos). El motivo de esto es que crear distintas clases monitores para proteger las distintas tablas del servidor nos proporciona exclusión mutua en las operaciones que realizamos con estas tablas de forma individual. Sin embargo, si es el servidor el que controla la concurrencia podemos proteger de concurrencia bloques enteros de operaciones sobre varias tablas.

Por ejemplo, para añadir un usuario modificamos varias tablas. Si hacemos cada modificación de estas tablas de forma independiente (a pesar de protegerlas con exclusión mutua), otras peticiones de otros usuarios podrían aceptarse y responderse ¡en estados internos inconsistentes del servidor! (a mitad de añadir un usuario).