Aix-Marseille Université
Application Réseaux
2022

TD-5 : Programmation Sockets avec Java NIO

A Channels et Buffers

1. Comment assurer la communication non-bloquante pour la lecture ou l'écriture dans un `SocketChannel` en Java NIO ? Quelles propriétés ont les objets de type `SelectableChannel`
2. Supposons qu'on ai un objet `buffer` de type `ByteBuffer` ayant pour valeurs internes : $Position = 1$, $Limit = 3$ et $Capacity = 8$. On exécute ensuite

   ```
   socketChannel.write(buffer);
   ```

   Combien de 'bytes' vont être transférés de `buffer` vers `socketChannel` ? Quelles seront les nouvelles valeurs de *Position*, *Limit* et *Capacity* ?
3. Quelles sont les différence entre les méthodes `Buffer.clear()` et `Buffer.flip()` pour un objet de type `ByteBuffer`. Expliquer avec un exemple précis.

B Java NIO - Classe `Selector`

1. Décrire la classe `Selector` et la classe `SelectionKey`. Combien d'objets de type `SelectionKey` peuvent être associes à un `Selector` ?
2. Comment pourriez-vous enregistrer un `SocketChannel` à un `Selector` pour attendre les événements de types écriture (WRITE) ? Comment pourriez-vous enregistrer un `ServerSocketChannel` à un `Selector` pour attendre les demandes de connexion des clients ?
3. On considère le code suivant pour la création d'un serveur et la connexion avec un premier client :

   ```
   ServerSocketChannel ssc = ServerSocketChannel.open();
   ssc.socket().bind(new InetSocketAddress(port));
   SocketChannel csc = ssc.accept();
   ```

   Créer un seul `Selector` qui attend deux types d'événements : (1) les demandes de connexion des clients vers le serveur (`ssc`). (2) les événements de types écriture par le client (`csc`).

C Concurrence en Java NIO

1. Est-il possible d'utiliser un seul objet `Selector` partagé par plusieurs threads ?
2. Quelle est l'utilité de la méthode `Selector.wakeup()` ? Quand avons-nous besoin de l'utiliser ?
3. Les méthodes pour la classe `Buffer`, sont ils "thread-safe" (possible de les utiliser de manière concurrente) ? Si plusieurs threads partagent le même `Buffer`, que passe-t-il ?
4. Avec Java NIO, un seul `Channel` peut utiliser plusieurs objets `Buffer` pour la communication. Quelles sont les classes et méthodes à utiliser pour une communication (1) d'un `Channel` vers plusieurs `Buffers`, et (2) de multiples `Buffers` vers un seul `Channel`.