

# TP1 - Remote Method Invocation

Réseaux Middlewares - Jules Chevalier

Le but de ce TP est de mettre en place un serveur et un client RMI. Pour commencer, le client et le serveur seront sur la même machine, puis vous ferez communiquer des clients et serveurs sur différentes machines. Vous implémenterez une petite application réseau utilisant RMI pour communiquer.

## 1 Echo à distance

### 1.1 Local

L'objectif est d'implémenter l'exemple du cours, à savoir un Echo accessible à distance grâce au RMI. Pour rappel, vous devrez écrire 4 classes :

- **RemoteEcho** : L'interface qui définit les méthodes à implémenter
- **RemoteEchoImpl** : L'implémentation de cette interface
- **Server** : Le serveur qui initialise le registre RMI, crée une instance de **RemoteEchoImpl** et l'enregistre dans le registre
- **Client** : Le client, qui récupère l'objet distant grâce au registre RMI, et appelle les méthodes de l'objet

L'interface RemoteEcho doit être commune, puisqu'elle joue le rôle de "contrat". La voici :

```
1  import java .rmi .Remote;
2  import java .rmi .RemoteException;
3
4  public interface RemoteEcho extends Remote {
5
6      String hello () throws RemoteException;
7      String echo(String s) throws RemoteException;
8
9  }
```

L'implémentation de cette interface est libre. Le but est que chacun ait une implémentation différente, pour la phase suivante. La méthode **hello()** doit renvoyer une chaîne de caractère du type "Bonjour". La méthode **Echo(String s)** doit quand à elle renvoyer une chaîne de caractère **s** modifiée, par exemple "Le message est : " suivi de **s**.

Attention, la classe **RemoteEchoImpl** doit étendre la classe **UnicastRemoteObject** pour fonctionner, et chacune de ses méthodes doit lever l'exception **RemoteException**.

Pour vous aider, voici quelques lignes de code :

1. Création du registre RMI :

```
1 Registry registry = LocateRegistry . createRegistry ( port );
```

2. Enregistrement d'un objet dans le registre RMI :

```
1 registry . rebind( name, object );
```

3. Récupération du registre RMI (pour le client) :

```
1 Registry registry = LocateRegistry . getRegistry ( serverAdress , port );
```

4. Récupérer un objet du registre RMI

```
1 String s = ( String ) registry . lookup( name );
```

Pour commencer, utilisez l'adresse *localhost* pour le client.

Une fois toutes les classes écrites, vous devez lancer le serveur, puis lancer le client. Le client doit appeler et afficher le résultat des méthodes **hello** et **echo**.

## 1.2 Distant

Une fois que votre serveur et votre client échangent en local, remplacez l'adresse *localhost* par votre adresse ip, puis lancez une fois de plus le serveur et le client. Si ça fonctionne toujours, vous allez pouvoir faire communiquer vos serveur et vos clients.

En binome, essayez d'utiliser l'objet distant de l'autre dans votre client. Vous devez bien sur pour cela utiliser son adresse IP, et donc être sur le même réseau (câble ou WIFI).

Comme précisé dans le cours, RMI est un modèle sécurisé permettant de communiquer. Tant que la connexion est local, il n'y a aucun problème. Par contre, pour joindre un serveur situé sur une autre machine, vous devez définir des paramètres de sécurité. Pour des questions pratiques, nous allons définir une sécurité minimum, ce qui permettra à tout le monde de se connecter à votre serveur. Pour cela, créez à la racine de votre projet un fichier **server.policy** :

```
1 grant {
2     permission java . security . AllPermission ;
3 };
```

On définit ici que toutes les permissions sont ouvertes pour le serveur. Si vous finissez le TP en avance, vous pouvez chercher comment gérer les droits par IP.

Pour que le fichier soit pris en compte par le programme, vous devez le lancer avec les options suivantes :

```
1 -Djava . security . manager -Djava . security . policy = server . policy
```

## 2 Rendu

Le TP en lui-même n'est pas noté. Cependant, ceux qui le désire peuvent m'envoyer leur code (Echo à distance et/ou Annuaire), afin d'obtenir un *petit* bonus sur la moyenne du module.

Envoyez votre code au format **zip** ou **tar** par mail à [jules.chevalier@univ-st-etienne.fr](mailto:jules.chevalier@univ-st-etienne.fr), avec comme sujet "TP1 RMI nom1 nom2".