

BPO

TP Application Client-Serveur - Partie 2

À ce stade, l'application client-serveur fournit les fonctionnalités attendues : un client envoie un message à un serveur qui lui retourne une réponse. Nous allons maintenant structurer un peu plus l'application.

1) La création des couches UDP, IP et Ethernet de la pile de protocoles d'une machine se fait toujours sur le même modèle (cf. code écrit dans la classe **TestReseau**). Pour simplifier cette création, écrire la classe **reseau.Machine**, avec :

- **Machine (String nomMachine, Adresse adrIP, AdresseMac adrMac, Adresse masque)**
- **void ajouter (int port, Application appli)**
- **Liaison getCoucheLiaison()**

Corriger la classe **TestReseau** de façon à utiliser ce constructeur pour construire les deux machines et les connecter. Une application est créée pour une machine donnée.

Application appliClientNum = new ClientNumerique (portClient, machine)

Il est temps de créer un réseau de machines.

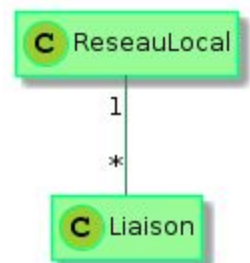
2) Pour simuler un réseau local, on complète le diagramme de classes avec la classe **reseau.ReseauLocal**.

- a) Écrire la définition de cette nouvelle classe avec au moins un constructeur **ReseauLocal()** et une fonction permettant d'ajouter une machine sur le réseau :

void ajouter (Machine m)

- b) Ajouter la fonction **setReseau (ReseauLocal rl)** dans la classe **Liaison**.

- c) Compléter la classe **TestReseau** pour créer un réseau local de plusieurs machines.



3) Pour que l'envoi d'un message se fasse correctement sur toutes les machines du réseau local, il faut corriger le texte de la fonction **sendMessage** de la classe **Ethernet** : la trame est dorénavant transmise au réseau local. Ajouter la fonction **void sendTrame(Message trame)** à la classe **ReseauLocal**. Vérifier que l'envoi d'un message se passe correctement.

Le schéma suivant synthétise le travail effectué dans ce TP : on a créé un réseau local avec deux machines. Sur la première machine, on a attaché une application cliente et sur la seconde deux applications serveur.

