

# Architecture d'applications Internet

François LEPAN

14 février 2013

## 1 Diffusion de messages

### 1.1 Quel peut-être l'avantage d'une diffusion en arbre par rapport à une diffusion habituelle (par exemple de type Multicast IP) ?

On évite la surcharge réseaux.

### 1.2 Écrire en Java le code d'un nœud intermédiaire (dans l'exemple 2 et 5 sont des nœuds intermédiaires)

```
public Class Noeud {
public void messageArrive(byte[] msg) {
transfererMessage(msg);
traiterMessage(msg);
}

void transfererMessage(byte[] msg) {
for (Noeud n : fils) {
envoyerMessage(adresse, msg);
}
}

void traiterMessage(byte[] msg) {
// do stuff
}

voir envoyerMessage(InetAddress adresse, byte[] msg) {
DatagramSocket ds = new DatagramSocket();
ds.send(new DatagramPacket(msg, adresse, 5000));
ds.close();
}

voir recevoirMessage() {
DatagramSocket ds = new DatagramSocket(5000);
byte[] msg = new byte[2048];
ds.receive( new DatagramPacket(msg));
ds.close();
}
```

```
this.messageArrive(msg);  
}  
}
```

## 2 Conception d'un protocole client/serveur et RPC

### 2.1 Le protocole est-il avec ou sans état ? Si la réponse est avec état, quel est l'état ? Sinon, pourquoi n'y a-t-il pas d'état ?

Non il n'y a pas d'état car il n'y a qu'un seul message et donc pas d'ordre d'envoi.

### 2.2 Plusieurs clients peuvent-ils transférer simultanément le même fichier ? Justifier.

Oui car plusieurs clients lisent le même message et de plus le protocole n'a pas d'état

### 2.3 Lors d'une demande de transfert de fichier par un client, quelle modification cette fonctionnalité entraîne-t-elle au niveau du protocole ? Que doit indiquer en plus le client ?

Il faut indiquer en plus le point de reprise du fichier. (ici  $100\text{Ko} * 8 * 1000 * 5 \text{ bits}$ )

### 2.4 Lors de l'envoi d'un message PRep, le serveur attend d'avoir reçu un acquittement (PRepAck) avant de continuer. Citer deux fonctionnalités qu'un tel mécanisme permet de réaliser.

waitMessage; ??

### 2.5 Le protocole est-il maintenant avec ou sans état ?

Le serveur va maintenant maintenir un état par connexion mais le protocole n'en a pas.

### 2.6 Dans le cas particulier où le fichier s'appelle fic et contient 225 Ko, indiquer sur un diagramme la séquence de messages échangés par le client et le serveur pour un transfert complet du fichier.

Voir Fig. 1

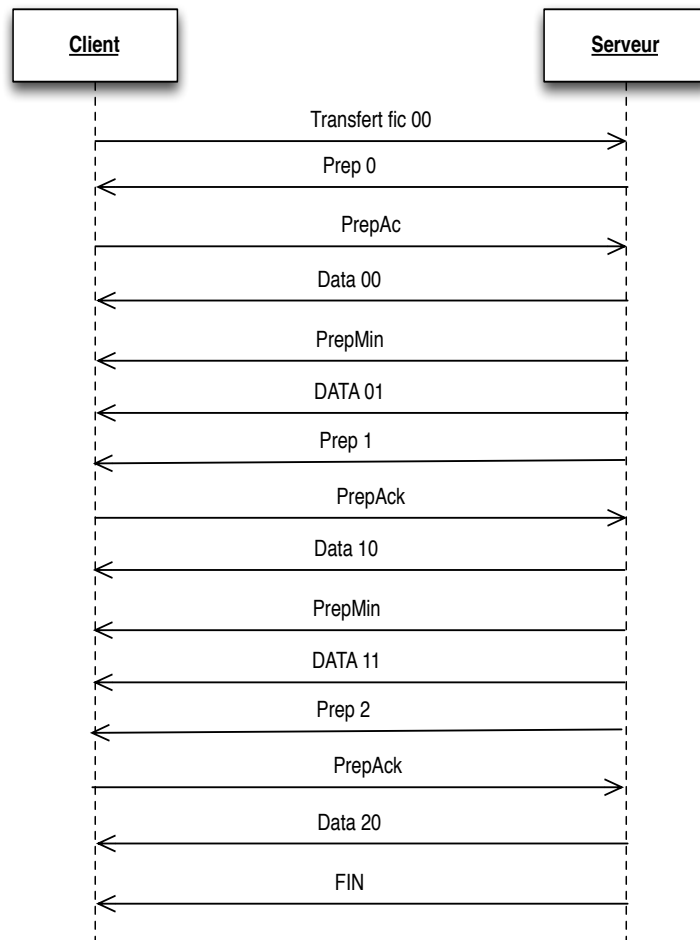


FIGURE 1 – Diagramme de séquence 2 - Q5

**2.7 En utilisant les primitives de la question précédente, écrire le pseudo-code ou le code Java des primitives `client(fic,i,j)` et `serveur()`**

```

client(fic,i,j) {
    String msg;
    OuvrirCx(Serveur);
    attendreCx();
    envoyer("TRANSFERT "+fic+ " "+i+ " "+j);

    processMess(mes);

    while ((mes = recevoir()) != "FIN") {

```

```

        if (msg == "Prep") {
            // stocker i;
            envoyer("PrepAck");
        } else if (msg == "PrepMin") {
            // stocker j
        } else if (msg == "DATA" ){
            // stocker dans n fichier
        }
    }

    fermerCx();
}

envoyerFichier(String fic) {
    int nb = nbBloc100K(fic);
    int prep = 0, j = 0;

    for (int i = 0; i < nb ; i++) {
        if (i%2 == 0) {
            envoyer("Prep" prep++);
            recevoir(); // pour ack
            j = 0
        }
        else {
            envoyer("PrepMin");
            j = 1;
        }
        envoyer ("DATA"+i+" "+j);
    }
}

```