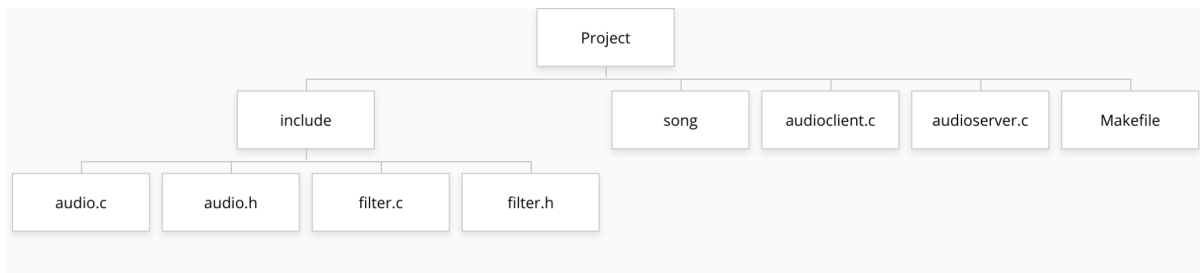


Compte rendu - Projet

Un système de streaming audio

FOUILLEN Mathis, GOMBERT Gwenaël

Arborescence du projet



Les fichiers audio (.wav) sont à déposer dans le dossier Project/song

Partie 1 : Streaming audio

A) Exécution et gestion des paquets perdus.

L'exécution du serveur se fait de la manière suivante :

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ ./audioServer
```

L'exécution du client se fait de la manière suivante :

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test
requested chans=2, sample rate=44100 sample size=16
set chans=2, sample rate=44100 sample size=16
```

**Les fichiers ont depuis renommés audioserver et audioclient après la rédaction de ce compte rendu.*

La connexion est en UDP.

Pendant l'écriture de la musique, un acquittement est fait entre le client et le serveur, pour s'assurer du bon fonctionnement de l'application.

Le Time To Live des paquets est de 2 secondes, après quoi ils sont considérés comme perdus (timeout). On a supposé que les paquets contenant les paramètres de la musique (sample_size, sample_rate et channels) ne sont jamais perdus.

Si un paquet est perdu (côté client ou serveur), la musique fait une légère pause avant de reprendre.

Si trois paquets d'affilés sont perdus (côté client ou serveur), la connexion est coupée.

B) Gestion de plusieurs clients et affichage des messages d'erreur.

Le serveur peut servir un ou plusieurs clients à la fois (on envoie à chaque nouveau client un nouveau numéro de port, sur lequel on fait un nouveau socket), il y a aussi une version en commentaire pour ne gérer qu'un client si on veut que notre serveur ne s'occupe que d'un seul client.

```
56
57         //remove the following comments to limit to a single client
58
59         /*
60         int stat;
61         while(pid=waitpid(-1, &stat, WNOHANG) == 0) { //while child alive
62             len = recvfrom(fd, buffer, SIZE, 0, (struct sockaddr*) &from, &flen);
63             sendto(fd, "server in use", SIZE, 0, (struct sockaddr*) &from, sizeof(struct sockaddr_in));
64         }
65         */
```

Extrait du code audioServer.c

Dans la version avec un seul client, si un autre client se connecte il reçoit (et affiche) un message d'erreur. Une fois la connexion avec le premier client terminée, un autre client peut se connecter sans avoir besoin de relancer le serveur.

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test
Server already used
prgc@prgc-VirtualBox:~/sys/projet/serveur$
```

Partie 2 : Filtres

A) Les filtres mono, volume et echo

Les filtres sont codés dans le fichier include/filter.c

Trois filtres sont disponibles, mono, qui transfère un signal mono même si le fichier est un stéréo, volumeEdit qui augmente ou diminue le son, et echo qui ajoute de l'écho, l'interface est la suivante :

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test mono
requested chans=1, sample rate=44100 sample size=16
set chans=1, sample rate=44100 sample size=16
^C
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test echo
requested chans=2, sample rate=44100 sample size=16
set chans=2, sample rate=44100 sample size=16
^C
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test volume 30
requested chans=2, sample rate=44100 sample size=16
set chans=2, sample rate=44100 sample size=16
^C
prgc@prgc-VirtualBox:~/sys/projet/serveur$ █
```

Pour le filtre volume, on entre derrière le mot clé, une valeur (si aucune valeur n'est écrite, le filtre n'aura aucun effet).

Pour une musique de sample_size 16 la valeur peut aller de 0.1 à 20.

Pour une musique de sample_size 8 la valeur peut aller de 0.3 à 1.8.

Si une valeur incorrecte est rentrée, cette dernière est ajustée pour le bon fonctionnement de l'application.

B) Plusieurs filtres simultanément

Il est possible de combiner tous les filtres en même temps :

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test mono echo  
requested chans=1, sample rate=44100 sample size=16  
set chans=1, sample rate=44100 sample size=16
```

```
prgc@prgc-VirtualBox:~/sys/projet/serveur$ padsp ./audioClient test mono echo volume 0.8  
requested chans=1, sample rate=44100 sample size=16  
set chans=1, sample rate=44100 sample size=16
```

Attention : Appliquer le filtre echo et le filtre mono simultanément semble détériorer le son.