

QjackCtl

QjackCtl est une application Qt simple pour contrôler le démon de serveur de son JACK, spécifique à l'infrastructure Linux Audio Desktop.

Écrit en C++ autour du framework Qt pour X11, en utilisant exclusivement Qt Designer.

Fournit un simple dialogue GUI pour définir plusieurs paramètres du démon JACK, qui sont correctement sauvegardés entre les sessions, et un moyen de contrôler l'état du démon du serveur audio. Avec le temps, cette interface primordiale s'est enrichie en incluant un patchbay amélioré et des fonctions de contrôle des connexions.

Installation

Il faut installer les paquets `qjackctl` et `pulseaudio-module-jack`

```
sudo apt-get install qjackctl pulseaudio-module-jack
```

Il faut ensuite ajouter l'utilisateur au groupe **audio**

```
sudo usermod -aG audio $USER
```

Une fois fait, pour mon utilisation, je crée des entrées, sorties son virtuel en plaçant un script dans mon dossier personnel, je l'ai mis dans `~/qjackctl/script-demarrage.sh` Voici un exemple du contenu de mon script :

```
#!/bin/bash
#BRIDGE JACK out <----> PULSEAUDIO in
#BRIDGE JACK in <----> PULSEAUDIO out
#
# WHY ? and why not?!!! because Discord only support Pulseaudio
#
# because Firefox binary arch are not compiled with jack support
bridge1="Discord"
bridge2="OBS"
bridge3="Son-system"
bridge4="autre"
# DISCORD BRIDGE PULSEAUDIO - JACK
pacmd load-module module-jack-source client_name=$bridge1 channels=2
pacmd load-module module-jack-sink client_name=$bridge1 channels=2
```

```
# OBS BRIDGE PULSEAUDIO - JACK
```

```
pacmd load-module module-jack-source client_name=$bridge2 channels=2
```

```
pacmd load-module module-jack-sink client_name=$bridge2 channels=2
```

```
# SON-SYSTEM BRIDGE PULSEAUDIO - JACK
```

```
pacmd load-module module-jack-source client_name=$bridge3 channels=2
```

```
pacmd load-module module-jack-sink client_name=$bridge3 channels=2
```

```
# AUTRE BRIDGE PULSEAUDIO - JACK
```

```
pacmd load-module module-jack-source client_name=$bridge4 channels=2
```

```
pacmd load-module module-jack-sink client_name=$bridge4 channels=2
```

Le chemin du script sera à préciser dans qjackctl dans **Réglages...** => **Options** => **Exécuter un script après le démarrage**. Ne pas oublier de spécifier l'interface dans **Réglages...**

Revision #2

Created 6 January 2023 17:41:21 by Julien

Updated 11 January 2023 07:55:05 by Julien