

Musique Assistée sous Ordinateur sous Linux

Partie installation

V1.2 du 11 janvier 2017

Par Olivier Hoarau (olivier.hoarau@funix.org)

Table des matières

1 Historique du document.....	3
2 Préambule et licence.....	3
3 Avertissement.....	3
4 Présentation.....	4
5 Installation de bibliothèques.....	6
5.1 LV2.....	6
5.2 suil.....	6
5.3 Serd.....	7
5.4 Sord.....	7
5.5 Portaudio.....	7
5.6 Aubio.....	8
5.7 LADSPA.....	8
5.8 Sratom.....	11
5.9 Lilv.....	11
5.10 Rubber band.....	12
5.11 DSSI.....	12
6 Utilitaires divers.....	12
6.1 JACK.....	12
6.2 QjackCtl.....	13
6.3 Patchage.....	13
6.4 Audacity.....	15
6.5 Hydrogen.....	17
6.6 Mixxxx.....	20
7 Installation des plugins VST.....	24
7.1 Computer Music Toolkit.....	25
7.2 SWH.....	25
7.3 Tom's Audio Processing LADSPA plugins.....	26
8 MIDI.....	26
8.1 Présentation MIDI.....	26
8.2 SoundFont.....	26
8.3 FluidSynth.....	26
8.4 Qsynth.....	27
8.5 Instruments MIDI.....	29
9 Configuration système.....	29
10 Outils de MAO.....	32
10.1 MusE.....	32
10.2 Linux MultiMedia Studio.....	33
11 ardour.....	35
12 Qtractor.....	39
13 Rosegarden.....	40

1 Historique du document

11/01/17	Passage à LV2 1.14.0, portaudio v190600, aubio 0.4.3, qjackctl 0.4.4, hydrogen 0.9.7, audacity 2.1.2, Mixxx 2.0.0, qsynth 0.4.3, Ardour 5.5.0 et rosegarden 16.06
19/07/15	Document scindé en une partie installation et une partie utilisation
20/03/15	Création du document

2 Préambule et licence

Ce document présente l'installation des outils utiles pour faire de la musique assistée sous ordinateur (MAO) sous Linux.

La dernière version de ce document est téléchargeable à l'URL <http://www.funix.org>.

Ce document est sous licence Creative Commons Attribution-ShareAlike 3.0 Unported, le détail de la licence se trouve sur le site <http://creativecommons.org/licenses/by-sa/3.0/legalcode>. Pour résumer, vous êtes libres

- de reproduire, distribuer et communiquer cette création au public
- de modifier cette création

suivant les conditions suivantes:

- **Paternité** — Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'oeuvre).
- **Partage des Conditions Initiales à l'Identique** — Si vous transformez ou modifiez cette oeuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

Par ailleurs ce document ne peut pas être utilisé dans un but commercial sans le consentement de son auteur. Ce document vous est fourni "dans l'état" sans aucune garantie de toute sorte, l'auteur ne saurait être tenu responsable des quelconques misères qui pourraient vous arriver lors des manipulations décrites dans ce document.

3 Avertissement

Le but de ce document est de vous présenter l'installation d'un ensemble d'outils vous permettant de faire de la musique assistée sous ordinateur sous Linux.

Ce document se veut le plus didactique possible pour qu'il puisse être utile à un maximum de personnes, novices comme gourous ! Cependant il ne prétend être un livre de recette qu'on suit à la lettre. Dans la plupart des cas ce sera le cas, mais il m'a parfois été nécessaire de réaliser certaines opérations, parfois peu orthodoxes (modification du code), pour pouvoir installer tel ou tel logiciel. En fonction de votre configuration logicielle, vous ne rencontrerez peut être pas de difficultés ou, au contraire, en rencontrerez là où je n'en ai pas eu.

J'ai essayé de détailler au possible les opérations réalisées pour arriver à compiler un logiciel, cela vous sera utile si vous rencontrez les mêmes difficultés ou si ce n'est pas le cas pour comprendre la

méthodologie à adopter pour résoudre un problème de compilation.

Les machines de test utilisées pour écrire ce document tourne avec une distribution Mageia. Ce document est donc orienté Mageia, mais il s'applique aussi à d'autres distributions étant donné que je présente l'installation de logiciels en utilisant uniquement les sources et que l'arborescence et les fichiers système varient peu d'une distribution à une autre.

Si vous éprouvez des difficultés de compréhension, rencontrez des problèmes de compilation inattendus, ou d'une manière générale si vous avez une remarque qui pourrait contribuer à améliorer ce document, n'hésitez pas à en faire part à l'auteur (olivier.hoarau@funix.org).

4 Présentation

Ce chapitre a pour objet de présenter l'installation des outils principaux permettant de faire de la musique assistée par ordinateur (MAO).

Je dispose de deux configurations matérielles type:

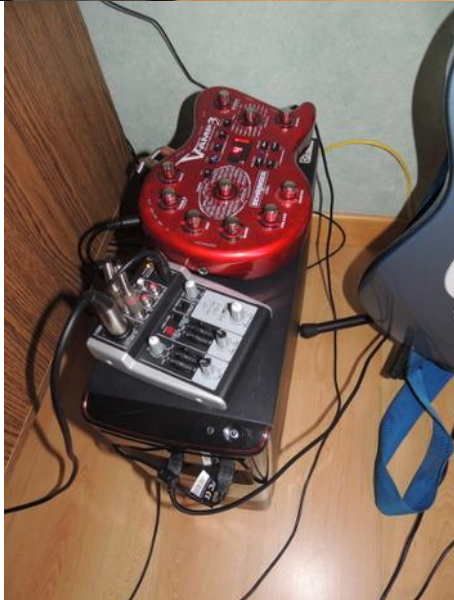
- une configuration mobile pour ordinateur portable avec une interface USB audio [Behringer UCA222](#) sur laquelle est branchée un boîtier d'effets pour guitare [Behringer V-AMP3](#) sur lequel est connectée une guitare électrique, le son est renvoyé vers les enceintes du PC.

- une configuration fixe avec une mini table de mixage [Behringer Xenyx 302 USB](#) sur laquelle est branchée un boîtier d'effets pour guitare [Behringer V-AMP3](#) sur lequel est connectée une guitare électrique, un micro d'enregistrement et un piano numérique

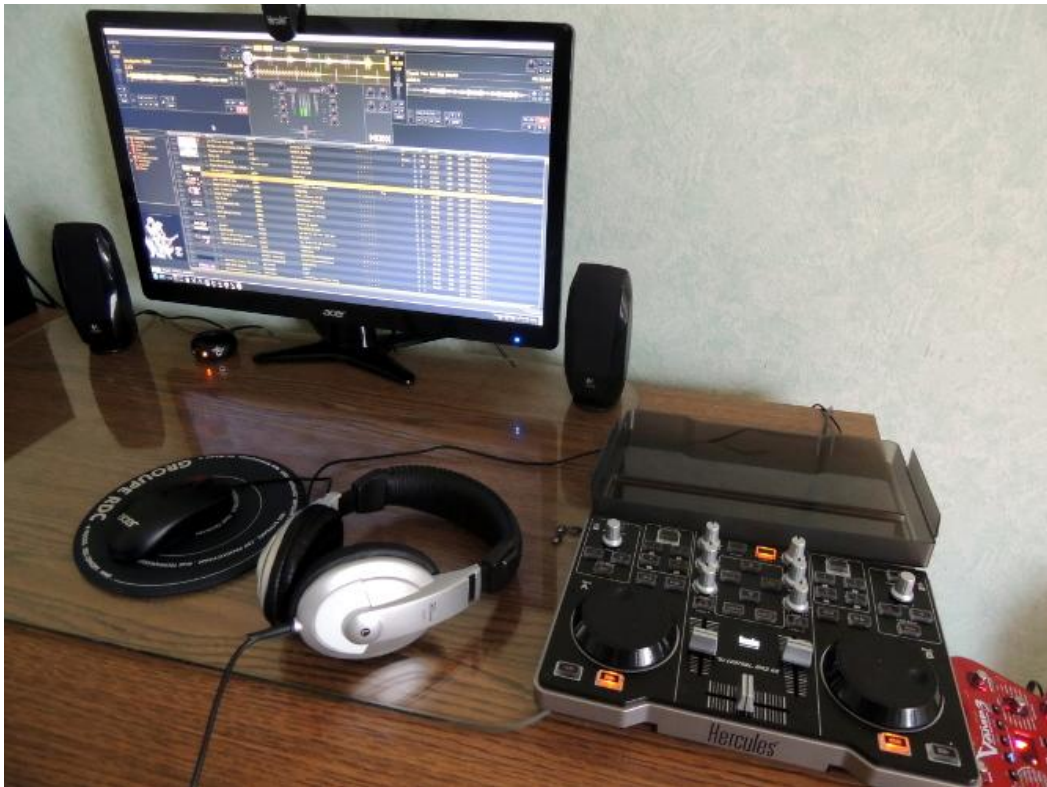
Quelle que soit la configuration, le son est renvoyé sur les enceintes du PC. Les éléments de configuration qui suivront seront basés sur ces matériels

Je dispose également d'une table de contrôle de mixage [Hercules DJ Control MP3 e2](#) que j'utilise avec **mixxx**, ce n'est pas vraiment un outil de MAO, mais il permet quand même de faire de la création et d'enregistrer ses œuvres. C'est pour ça qu'on le retrouve ici.

Sur la photo ci-dessous ma configuration fixe de mon modeste "studio".



Un focus sur mes interfaces xenyx 302 et Behringer V-AMP3



Un autre sur la table de DJ Hercules avec **mixxx**.

5 Installation de bibliothèques

Certaines bibliothèques logicielles sont obligatoires pour pouvoir utiliser des logiciels de plus haut niveau. Leur installation est détaillée ci-dessous.

5.1 LV2

La bibliothèque LV2 est un standard de plugin pour les systèmes audio, c'est grâce à cette bibliothèque qu'on peut intégrer des plugins compatibles LV2 dans les outils de MAO de haut niveau. On installera préalablement les packages **lib64sndfile-progs** et **lib64sndfile-devel**. Le site officiel est <http://lv2plug.in> on y récupère l'archive qu'on décompresse en tapant :

```
tar xvfj lv2-1.14.0.tar.bz2
```

cela donne le répertoire **lv2-1.14.0** dans lequel on tape successivement

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.2 suil

suil est une bibliothèque pour gérer les plugins LV2. Le site officiel est

<http://drobilla.net/software/suil/> on y récupère l'archive qu'on décompresse en tapant

```
tar xvfj suil-0.8.2.tar.bz2
```

cela donne le répertoire **suil-0.8.2** dans lequel on tape

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.3 *Serd*

Serd est une bibliothèque C fournissant des routines pour la syntaxe RDF ([Resource Description Framework](#)) pour des logiciels disposant de ressources limitées ou nécessitant des performances élevées. Le site internet est <http://drobilla.net/software/serd/> on y récupèrera l'archive qu'on décompressera en tapant

```
tar xvfj serd-0.24.0.tar.bz2
```

cela donne le répertoire **serd-0.24.0** dans lequel on tape successivement

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.4 *Sord*

Sord est une bibliothèque C qui fournit des routines pour stocker des données RDF en mémoire. Le site officiel est <http://drobilla.net/software/sord/> on y récupèrera l'archive qu'on décompresse en tapant :

```
tar xvfj sord-0.16.0.tar.bz2
```

cela donne le répertoire **sord-0.16.0** dans lequel on tape successivement

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.5 *Portaudio*

Portaudio est une bibliothèque qui gère les entrées sorties audio. Le site officiel est <http://www.portaudio.com> on y récupère la dernière version qu'on décompresse en tapant

```
tar xvfz pa_stable_v190600_20161030.tgz
```

cela donne le répertoire **portaudio** dans lequel on tape

```
./configure --with-jack  
make
```

puis en tant que root

```
make install
```

5.6 *Aubio*

Aubio est un outil qui permet d'extraire des annotations d'un signal audio. Le site officiel est <http://aubio.org/> on y récupère la dernière version qu'on décompresse en tapant

```
tar xvfj aubio-0.4.3.tar.bz2
```

cela donne le répertoire **aubio-0.4.3** dans lequel on tape successivement

```
./waf configure  
./waf
```

puis en tant que root

```
./waf install
```

5.7 *LADSPA*

A l'instar de **LV2** Linux Audio Developer's Simple Plugin API (**LADSPA**) est une bibliothèque fournissant des routines pour pouvoir intégrer des plugins audio. **LADSPA** est maintenant obsolète et est supplanté par le format **LV2**. Le site officiel est <http://www.ladspa.org/> il est down maintenant, comme beaucoup de softs se basent encore sur **LADSPA** vous pourrez récupérer [ici](#) les sources qu'on décompresse en tapant

```
tar xvfz ladspa_sdk_1.13.tgz
```

cela donne le répertoire **ladspa_sdk** dans lequel on va éditer le fichier **src/makefile** pour modifier ainsi les lignes suivantes

```
INSTALL_PLUGINS_DIR = /usr/lib64/ladspa/
```

et

```
-mkdir $(INSTALL_PLUGINS_DIR)  
-mkdir $(INSTALL_INCLUDE_DIR)  
-mkdir $(INSTALL_BINARY_DIR)
```

on revient sous **src** on tape **make** puis en tant que root **make install**

Une fois les [plugins VST](#) installés, vous disposez de la commande **listplugins** qui comme son nom l'indique va liste les plugins compatibles **LADSPA** installés sur le système. Voilà le résultat avec

les plugins présentés dans cette page.

/usr/lib64/ladspa/delay.so:
Simple Delay Line (1043/delay_5s)

/usr/lib64/ladspa/tap_reflector.so:
TAP Reflector (2154/tap_reflector)

/usr/lib64/ladspa/tap_pinknoise.so:
TAP Pink/Fractal Noise (2155/tap_pinknoise)

/usr/lib64/ladspa/tap_echo.so:
TAP Stereo Echo (2143/tap_stereo_echo)

/usr/lib64/ladspa/tap_doubler.so:
TAP Fractal Doubler (2156/tap_doubler)

/usr/lib64/ladspa/tap_dynamics_st.so:
TAP Dynamics (St) (2153/tap_dynamics_st)

/usr/lib64/ladspa/ladspa-rubberband.so:
Rubber Band Mono Pitch Shifter (2979/rubberband-pitchshifter-mono)
Rubber Band Stereo Pitch Shifter (9792/rubberband-pitchshifter-stereo)

/usr/lib64/ladspa/tap_rotspeak.so:
TAP Rotary Speaker (2149/tap_rotspeak)

/usr/lib64/ladspa/sine.so:
Sine Oscillator (Freq:audio, Amp:audio) (1044/sine_faaa)
Sine Oscillator (Freq:audio, Amp:control) (1045/sine_faac)
Sine Oscillator (Freq:control, Amp:audio) (1046/sine_fcaa)
Sine Oscillator (Freq:control, Amp:control) (1047/sine_fcac)

/usr/lib64/ladspa/filter.so:
Simple Low Pass Filter (1041/lpf)
Simple High Pass Filter (1042/hpf)

/usr/lib64/ladspa/tap_chorusflanger.so:
TAP Chorus/Flanger (2159/tap_chorusflanger)

/usr/lib64/ladspa/noise.so:
White Noise Source (1050/noise_white)

/usr/lib64/ladspa/tap_eqbw.so:
TAP Equalizer/BW (2151/tap_equalizer_bw)

/usr/lib64/ladspa/tap_pitch.so:
TAP Pitch Shifter (2150/tap_pitch)

/usr/lib64/ladspa/tap_reverb.so:
TAP Reverberator (2142/tap_reverb)

/usr/lib64/ladspa/tap_sigmoid.so:
TAP Sigmoid Booster (2157/tap_sigmoid)

/usr/lib64/ladspa/tap_dynamics_m.so:
TAP Dynamics (M) (2152/tap_dynamics_m)

/usr/lib64/ladspa/tap_autopan.so:
TAP AutoPanner (2146/tap_autopan)

/usr/lib64/ladspa/cmt.so:
Ambisonic Decoder (B-Format to Cube) (1092/bf2cube)
Ambisonic Decoder (B-Format to Quad) (1091/bf2quad)
Ambisonic Decoder (B-Format to Stereo) (1090/bf2stereo)
Ambisonic Decoder (FMH-Format to Octagon) (1093/fmh2oct)
Ambisonic Encoder (B-Format) (1087/encode_bformat)
Ambisonic Encoder (FMH-Format) (1088/encode_fmh)
Ambisonic Rotation (B-Format, Horizontal) (1094/bf_rotate_z)
Ambisonic Rotation (FMH-Format, Horizontal) (1095/fmh_rotate_z)
Amplifier (Mono) (1067/amp_mono)

Amplifier (Stereo) (1068/amp_stereo)
Amplitude Modulator (1070/am)
Analogue Voice (1221/analogue)
Canyon Delay (1225/canyon_delay)
Disintegrator (1846/disintegrator)
Dynamic Sledgehammer (1848/sledgehammer)
Echo Delay Line (Maximum Delay 0.01s) (1053/delay_0.01s)
Echo Delay Line (Maximum Delay 0.1s) (1054/delay_0.1s)
Echo Delay Line (Maximum Delay 1s) (1055/delay_1s)
Echo Delay Line (Maximum Delay 5s) (1056/delay_5s)
Echo Delay Line (Maximum Delay 60s) (1057/delay_60s)
Envelope Tracker (Maximum Peak) (1080/track_max_peak)
Envelope Tracker (Maximum RMS) (1081/track_max_rms)
Envelope Tracker (Peak) (1078/track_peak)
Envelope Tracker (RMS) (1079/track_rms)
FMH-Format to B-Format (Discards RSTUV Channels) (1089/fmh2bf)
Feedback Delay Line (Maximum Delay 0.01s) (1058/fbdelay_0.01s)
Feedback Delay Line (Maximum Delay 0.1s) (1059/fbdelay_0.1s)
Feedback Delay Line (Maximum Delay 1s) (1060/fbdelay_1s)
Feedback Delay Line (Maximum Delay 5s) (1061/fbdelay_5s)
Feedback Delay Line (Maximum Delay 60s) (1062/fbdelay_60s)
Freeverb (Version 3) (1123/freeverb3)
Granular Scatter Processor (1096/grain_scatter)
Hard Gate (1845/hard_gate)
High Pass Filter (One Pole) (1052/hpf)
Identity (Audio) (1098/identity_audio)
Identity (Control) (1099/identity_control)
Lo Fi (1227/lofi)
Logistic Map Control Generator (1849/logistic)
Low Pass Filter (One Pole) (1051/lpf)
Mixer (Stereo to Mono) (1071/mixer)
Noise Source (White) (1069/noise_source_white)
Null (Audio Input) (1084/null_ai)
Null (Audio Output) (1086/null_ao)
Null (Control Input) (1083/null_ci)
Null (Control Output) (1085/null_co)
Organ (1222/organ)
Peak Monitor (1082/peak)
Phase Modulated Voice (1226/phasemod)
Pink Noise (Interpolated) (1841/pink_interpolated_audio)
Pink Noise (full frequency range) (1844/pink_full_frequency)
Pink Noise (sample and hold) (1843/pink_sh)
Simple Compressor (Peak Envelope Tracking) (1072/compress_peak)
Simple Compressor (RMS Envelope Tracking) (1073/compress_rms)
Simple Expander (Peak Envelope Tracking) (1074/expand_peak)
Simple Expander (RMS Envelope Tracking) (1075/expand_rms)
Simple Limiter (Peak Envelope Tracking) (1076/limit_peak)
Simple Limiter (RMS Envelope Tracking) (1077/limit_rms)
Sine Oscillator (Freq:audio, Amp:audio) (1063/sine_faaa)
Sine Oscillator (Freq:audio, Amp:control) (1064/sine_faac)
Sine Oscillator (Freq:control, Amp:audio) (1065/sine_fcaa)
Sine Oscillator (Freq:control, Amp:control) (1066/sine_fcac)
Syn Drum (1223/syndrum)

```
VCF 303 (1224/vcf303)
Wave Shaper (Sine-Based) (1097/wshape_sine)
/usr/lib64/ladspa/tap_eq.so:
  TAP Equalizer (2141/tap_equalizer)
/usr/lib64/ladspa/amp.so:
  Mono Amplifier (1048/amp_mono)
  Stereo Amplifier (1049/amp_stereo)
/usr/lib64/ladspa/tap_limiter.so:
  TAP Scaling Limiter (2145/tap_limiter)
/usr/lib64/ladspa/tap_deesser.so:
  TAP DeEsser (2147/tap_deesser)
/usr/lib64/ladspa/tap_vibrato.so:
  TAP Vibrato (2148/tap_vibrato)
/usr/lib64/ladspa/tap_tubewarmth.so:
  TAP TubeWarmth (2158/tap_tubewarmth)
/usr/lib64/ladspa/tap_tremolo.so:
  TAP Tremolo (2144/tap_tremolo)
```

5.8 Sratom

Sratom est une bibliothèque C qui fournit des routines pour la bibliothèque LV2 en lien avec le protocole RDF. Le site officiel est <http://drobilla.net/software/sratom/> dans lequel on récupère l'archive qu'on décompresse en tapant

```
tar xvfj sratom-0.6.0.tar.bz2
```

cela donne le répertoire **sratom-0.6.0** dans lequel on tape successivement

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.9 Lilv

Lilv est une bibliothèque C qui rend l'intégration de plugins compatibles LV2 plus faciles pour les applications de MAO de haut niveau. Le site officiel est <http://drobilla.net/software/lilv/> on y récupèrera l'archive qu'on décompressera en tapant

```
tar xvfj lilv-0.24.2.tar.bz2
```

cela donne le répertoire **lilv-0.24.2** dans lequel on tape successivement

```
./waf configure
```

```
./waf
```

puis en tant que root

```
./waf install
```

5.10 Rubber band

Rubber Band Library est une bibliothèque pour réaliser des traitements audio, du genre changement de tempo. On installera préalablement le package suivant **lib64vamp-plugin-sdk-devel**. Le site officiel est <http://www.breakfastquay.com/rubberband/> dans lequel on récupèrera l'archive qu'on décompresse en tapant

```
tar xvfj rubberband-1.8.1.tar.bz2
```

cela donne le répertoire **rubberband-1.8.1** dans lequel on tape successivement

```
./configure
```

```
make
```

puis en tant que root

```
make install
```

5.11 DSSI

DSSI (à prononcer "dizzy") est une bibliothèque pour les plugins de traitement audio. Le site officiel est <http://dssi.sourceforge.net/> on y récupèrera l'archive qu'on décompresse en tapant

```
tar xvfz dssi-1.1.1.tar.gz
```

cela donne le répertoire **dssi-1.1.1** dans lequel on tape successivement

```
./configure
```

```
make
```

puis en tant que root

```
make install
```

on rajoutera au fichier **/etc/ld.so.conf** la ligne suivante

```
/usr/local/lib/dssi
```

on tape ensuite **ldconfig**

6 Utilitaires divers

6.1 JACK

JACK est un outil qui permet de gérer les flux audio temps réel avec peu de latence, il est indispensable à la plupart des outils de MAO. Le site officiel est <http://jackaudio.org/> dans lequel on récupèrera les sources qu'on décompresse en tapant

```
tar xvfj jack-1.9.10.tar.bz2
```

cela donne le répertoire **jack-1.9.10** dans lequel on tape successivement

```
./waf configure --alsa --portaudio
```

```
./waf build
```

puis en tant que root

```
./waf install
```

Attention de veiller à désinstaller tous packages de développement en lien avec **jack** en tapant **rpm -qa | grep jack** si vous obtenez quelque chose comme ça **lib64jack-devel-1.9.8-3.mga2** supprimez le avec **urpme**.

6.2 QjackCtl

QjackCtl est une interface graphique basée sur **Qt4** pour contrôler le serveur JACK. Le site officiel est <http://qjackctl.sourceforge.net/> on y récupèrera les sources qu'on décompresse en tapant

```
tar xvfz qjackctl-0.4.4.tar.gz
```

cela donne le répertoire **qjackctl-0.4.4** dans lequel on tape successivement

```
./configure --enable-jack-version
```

on rajoutera **--enable-qt4** si vous n'êtes pas encore dans un environnement **KF5** avec **Qt5**

```
make
```

puis en tant que root

```
make install
```

Et voilà ce que ça donne quand **qjackctl** est lancé avec le daemon **JACK** démarré



6.3 Patchage

patchage permet de visualiser et configurer les connexions audio entre les différents périphériques de la même manière que **qjackctl**. On installera préalablement la bibliothèque **ganv** qu'on trouvera par ici <http://drobilla.net/software/ganv> on y récupèrera l'archive qu'on décompresse en tapant

```
tar xvfj ganv-1.4.2.tar.bz2
```

cela donne le répertoire **ganv-1.4.2** dans lequel on tape

```
./waf configure
```

```
./waf
```

puis en tant que root

./waf install

on revient à **patchage** qu'on trouvera par ici <http://drobilla.net/software/patchage/> on y récupèrera l'archive qu'on décompresse en tapant

tar xvfj patchage-1.0.0.tar.bz2

cela donne le répertoire **patchage-1.0.0** dans lequel on tape

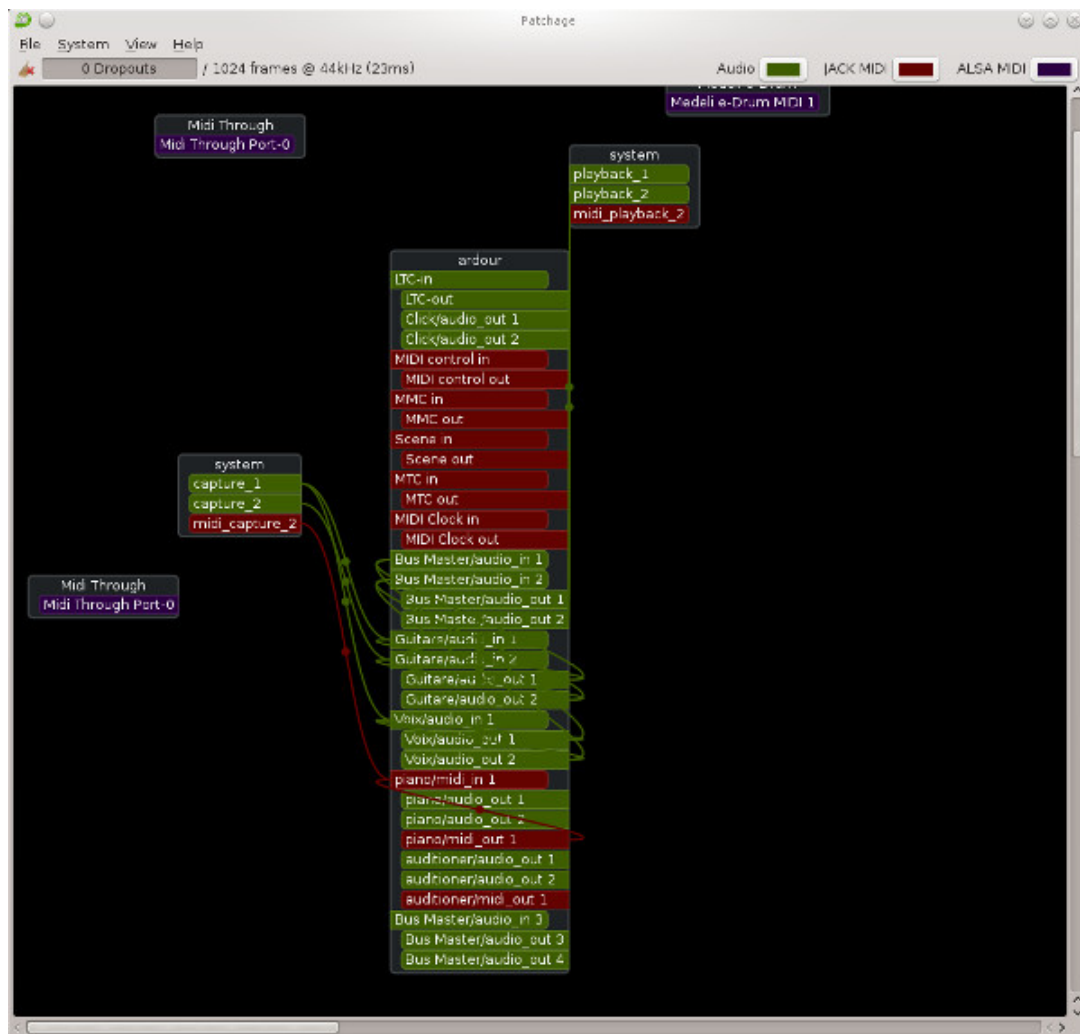
./waf configure

./waf

puis en tant que root

./waf install

voilà ce que ça donne



Les connexions se gèrent assez facilement par drag and drop, pour supprimer une connexion, il suffit de sélectionner une case puis avec le bouton droit de la souris **Disconnect**.

6.4 Audacity

Audacity est un outil d'édition audio qui permet de créer des bandes son, il permet également d'enregistrer à la partie de l'entrée micro. Le site officiel est <http://www.audacityteam.org/> où on récupérera la dernière version (instable) qu'on décompresse en tapant:

```
tar xvf audacity-minsrc-2.1.2.tar.xz
```

Cela donne le répertoire **audacity-minsrc-2.1.2**. Préalablement pour éviter l'erreur suivante

```
AudioIO.cpp:(.text+0x316c): undefined reference to `Resample::Resample(bool, double, double)'
```

on récupérera la bibliothèque **libsoxr** par là <http://sourceforge.net/p/soxr/wiki/Home/> on décompresse l'archive en tapant

```
tar xvf soxr-0.1.2-Source.tar.xz
```

cela donne le répertoire **soxr-0.1.2-Source/** dans lequel on tape

```
./go
```

puis en tant que root

```
cd Release; make install
```

On installera également les packages **libsndfile-devel** et **libid3tag-devel**. On revient sous **audacity-minsrc-2.1.2**

```
./configure --with-ffmpeg=local
```

audacity n'est pas vraiment compatible avec les dernières versions de **ffmpeg** d'où l'option pour utiliser la version compatible de **ffmpeg** fournie dans le package **audacity**. Voilà le résultat

Finished configure:

EXPAT: using SYSTEM libraries

FFMPEG: using LOCAL libraries

LAME: using SYSTEM libraries

LIBFLAC: using SYSTEM libraries

LIBID3TAG: using SYSTEM libraries

LIBMAD: disabled

LIBNYQUIST: using LOCAL libraries

LIBSBSMS: using LOCAL libraries

LIBSNDFILE: using SYSTEM libraries

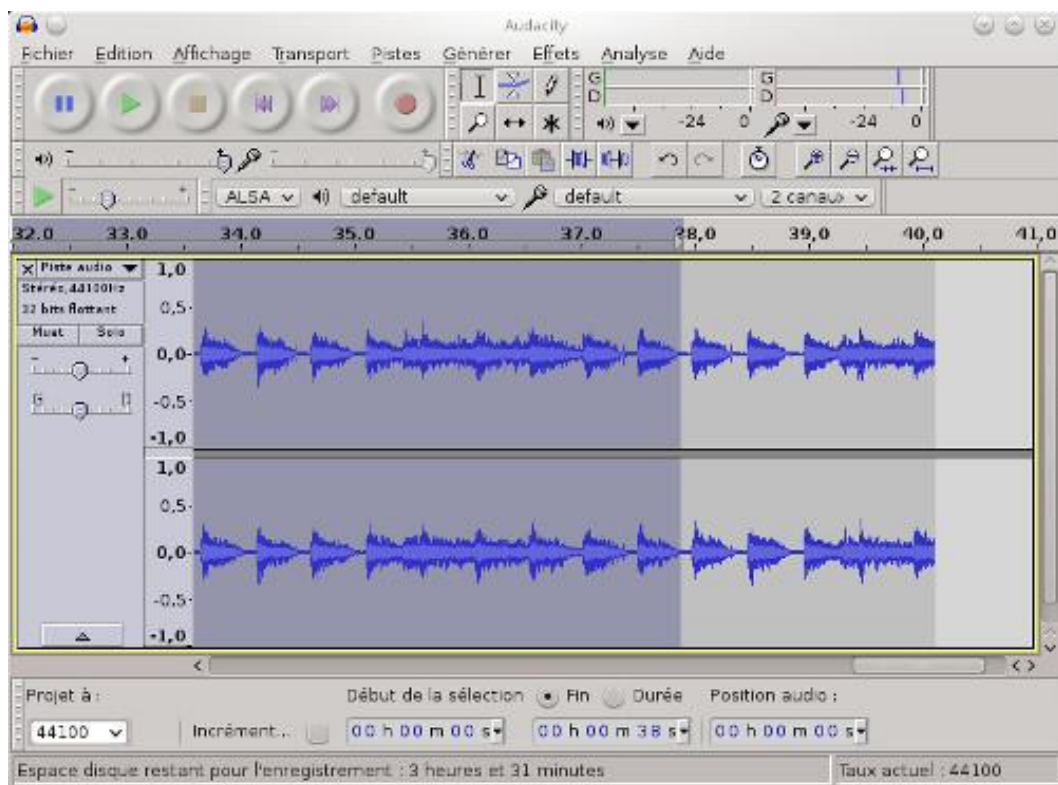
LIBSOUNDTOUCH: disabled

LIBSOXR: using SYSTEM libraries
LIBTWOLAME: using LOCAL libraries
LIBVAMP: using SYSTEM libraries
LIBVORBIS: using SYSTEM libraries
LV2: using SYSTEM libraries
PORTAUDIO: using LOCAL libraries
PORTSMF: using LOCAL libraries
WIDGETEXTRA: using LOCAL libraries
ladspa plugin support: enabled
audiounit plugin support: disabled
VST plugin support: enabled
prefix=/usr/local/

on tape alors **make** puis en tant que root

make install

voilà ce que ça donne



pour l'utilisation on trouve d'excellents tutoriels à ces adresses (en français)

www.bandits-mages.com/site2008/upload2/fichepeda_audacity.pdf

www.educlasse.ch/activites/coursinfo/documents/Audacity_tutoriel.pdf

<https://openclassrooms.com/courses/audacity-prise-en-main>

sinon toute une page de tutorial par ici <http://manual.audacityteam.org/#tutorials> (en anglais)

Quand vous voulez remplacer la bande son d'origine par une musique, l'idée est de préparer le mp3 sous **audacity**, ensuite le montage vidéo devra avoir exactement la même longueur en temps. Dans les effets audio de l'un des outils de montage vidéo cités plus haut, il faudra remplacer la bande son d'origine par votre fichier mp3 préparé.

si vous avez des soucis avec **audacity** en utilisant **pulseaudio**, il faudra lancer **audacity** en tapant

```
padsp audacity
```

ou

```
pasuspender -- audacity
```

6.5 Hydrogen

Pour faire simple **hydrogen** est une boîte à rythme , le site officiel est <http://www.hydrogen-music.org/hcms/> on y récupérera les sources qu'on décompresse en tapant

```
tar xvfz hydrogen-0.9.7.tar.gz
```

cela donne le répertoire **hydrogen-0.9.7** préalablement j'ai dû installer les packages **lib64tar-devel** et **lib64raptor-devel**, on tape ensuite

```
mkdir build
```

```
cd build
```

```
cmake ..
```

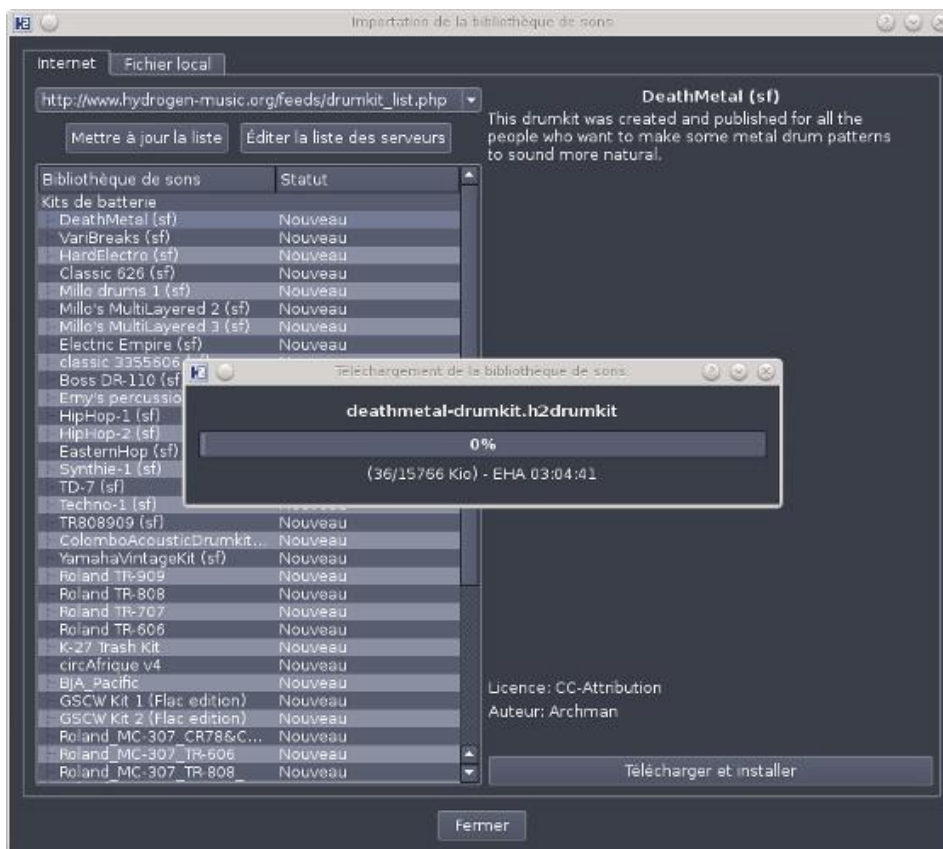
```
make
```

puis en tant que root

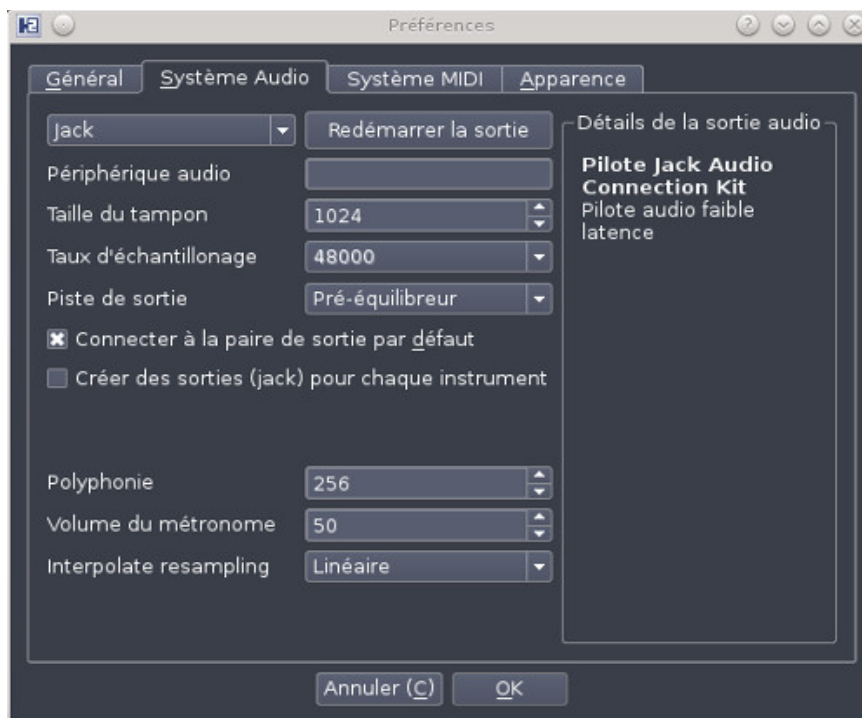
```
make install
```

Une fois **hydrogen** lancé on va rajouter une bibliothèque d'instruments à partir du menu **Instruments->Importer la bibliothèque**.

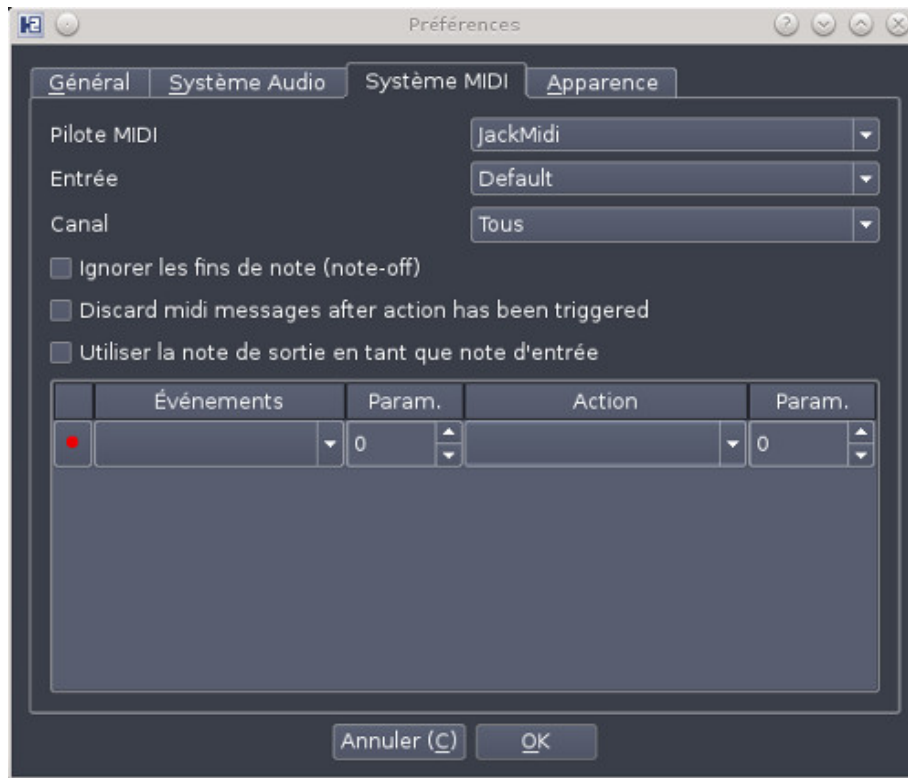
Il suffit ensuite de mettre à jour la liste puis de télécharger et installer les bibliothèques disponibles. Elles seront installées dans le répertoire utilisateur sous `~/.hydrogen/data/drumkits`. Pour que ces bibliothèques soient ensuite disponibles pour les autres utilisateurs, il faudra les copier sous `/usr/share/hydrogen/data/drumkits`



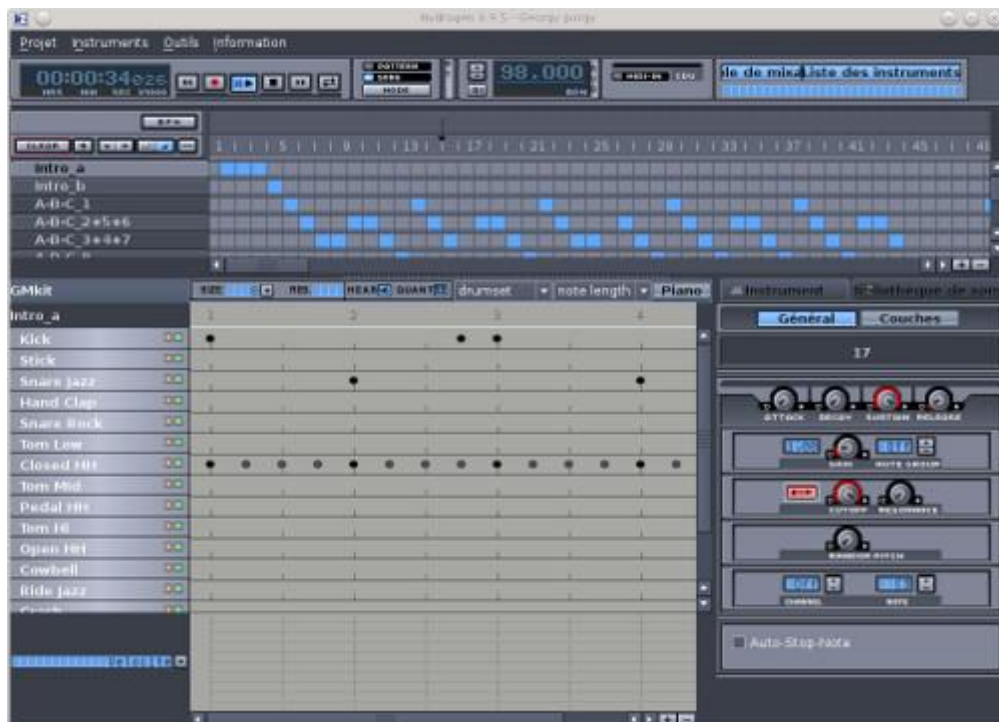
Maintenant dans les préférences, j'ai choisi **JACK**.



au niveau de l'onglet MIDI



Voilà à quoi ressemble **hydrogen** avec une démo chargée.



6.6 Mixxxx

Mixxxx est un outil de console DJ qui permet de gérer une liste de lecture en faisant des effets, chose intéressante il s'interface parfaitement avec la table Hercules DJ Control MP3 e2



Quand je la branche sur le port USB voilà ce que je peux voir dans le fichier `/var/log/messages`

```
Jan 12 13:54:02 mana kernel: [ 1564.628036] usb 2-3: new full-speed USB device number 4 using ohci_hcd
Jan 12 13:54:03 mana kernel: [ 1564.817049] usb 2-3: New USB device found, idVendor=06f8, idProduct=b105
Jan 12 13:54:03 mana kernel: [ 1564.817054] usb 2-3: New USB device strings: Mfr=1, Product=2, SerialNumber=0
Jan 12 13:54:03 mana kernel: [ 1564.817056] usb 2-3: Product: DJ Control MP3 e2
Jan 12 13:54:03 mana kernel: [ 1564.817058] usb 2-3: Manufacturer: Hercules
Jan 12 13:54:03 mana mtp-probe: checking bus 2, device 4:
"/sys/devices/pci0000:00/0000:00:02.0/usb2/2-3"
Jan 12 13:54:03 mana mtp-probe: bus: 2, device: 4 was not an MTP device
```

Le site officiel est <http://www.mixxxx.org>. Préalablement j'ai dû installer les packages suivants `portmidi-devel`, `lib64taglib-devel`, `protobuf-devel`, `lib64usb1.0-devel` et `lib64mp4v2-devel`. On décompresse l'archive en tapant

```
tar xvfz mixxxx-2.0.0-src.tar.gz
```

cela donne le répertoire `mixxxx-2.0.0` dans lequel on tape

```
scons shoutcast=0 faad=1 tuned=1
```

j'ai eu l'erreur suivante

```
src/musicbrainz/chromaprinter.cpp: In member function 'QString  
ChromaPrinter::calcFingerPrint(const SoundSourcePointer&):  
src/musicbrainz/chromaprinter.cpp:62:66: erreur: invalid conversion from 'void**' to  
'uint32_t** {aka unsigned int**}' [-fpermissive]
```

pour la résoudre on édite le fichier **build/features.py** et on rajoute **-fpermissive** à cette ligne

```
build.env.Append(CCFLAGS='-O3 -ffast-math -funroll-loops -fpermissive')
```

on retape **scons shoutcast=0 faad=1 tuned=1**

Puis en tant que root on tape

```
scons install
```

maintenant on lance **jackd** via **qjackctl** puis **mixxx** dans un shell, si vous voyez un message du genre

```
Debug [Controller]: Found output device # 0 Midi Through Port-0  
Debug [Controller]: Found input device # 1 Midi Through Port-0  
Debug [Controller]: Linking to output device # 0 "Midi Through Port-0"  
Debug [Controller]: Scanning USB Bulk devices:  
Debug [Controller]: Error opening a device  
Debug [Controller]: Scanning HID devices:  
Debug [Controller]: Found 0x0 0x0 "r2305" S/N 0x0 "Interface 0"  
Warning [Controller]: USB permissions problem (or device error.) Your account needs write  
access to USB HID controllers.  
Debug [Controller]: ControllerManager::getControllerList
```

c'est qu'il y a un problème de droit sur l'interface USB (vous pouvez faire un test sous root, ça devrait marcher). Dans ce cas, il faut créer le fichier **/etc/udev/rules.d/15-mixxx-usb.rules** qui contient

```
# Allow scanning USB devices
```

```
SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", GROUP="users"
```

```
# Allow communicating with HID devices
```

```
SUBSYSTEM=="usb", ATTR{bInterfaceClass}=="03", MODE="0660", GROUP="users"
```

faites en sorte que votre utilisateur soit dans le groupe **users**. Relancer **udev** (reboot), quand on relance **mixxx** on a maintenant le message suivant

Debug [Controller]: Searching for controller preset "DJ_Control_MP3_e2_.bulk.xml" in paths: "/home/olivier/.mixxx/controllers/,usr/local/share/mixxx/controllers/"

Debug [Controller]: Loading controller preset from "/home/olivier/.mixxx/controllers/DJ_Control_MP3_e2_.bulk.xml"

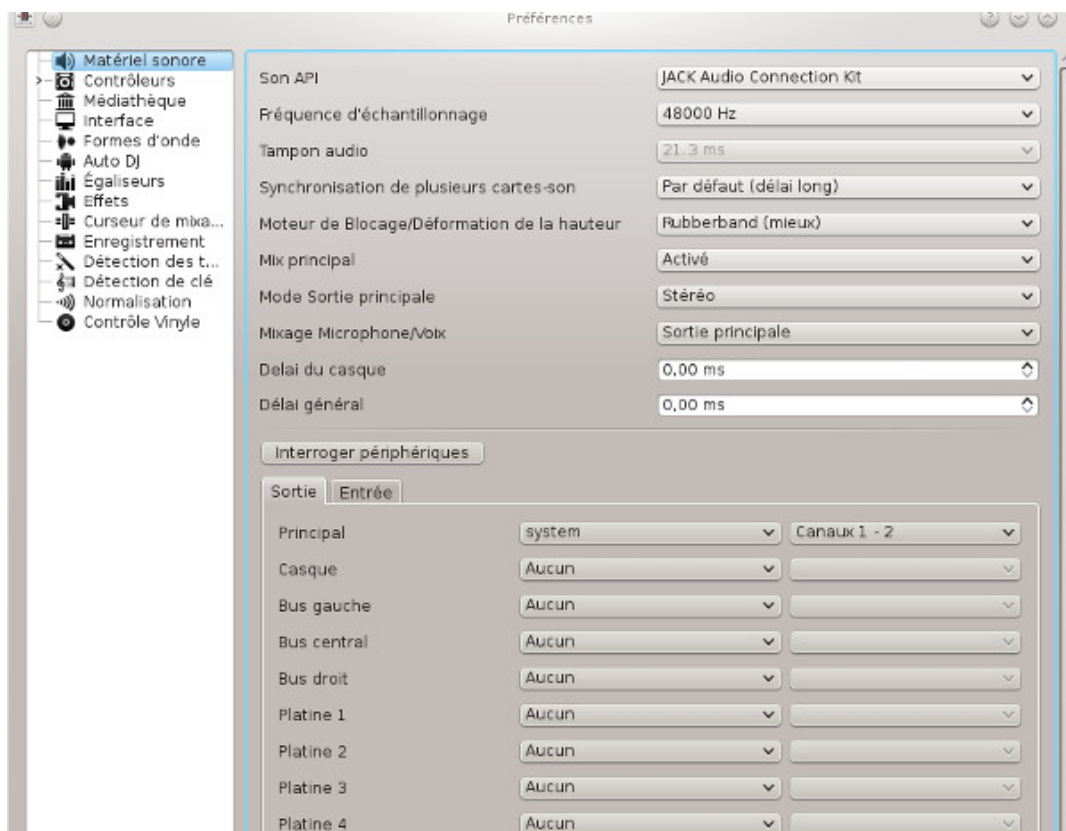
Debug [Controller]: Opening controller: "DJ Control MP3 e2 "

Debug [Controller]: Controller in script engine is: "DJ Control MP3 e2 "

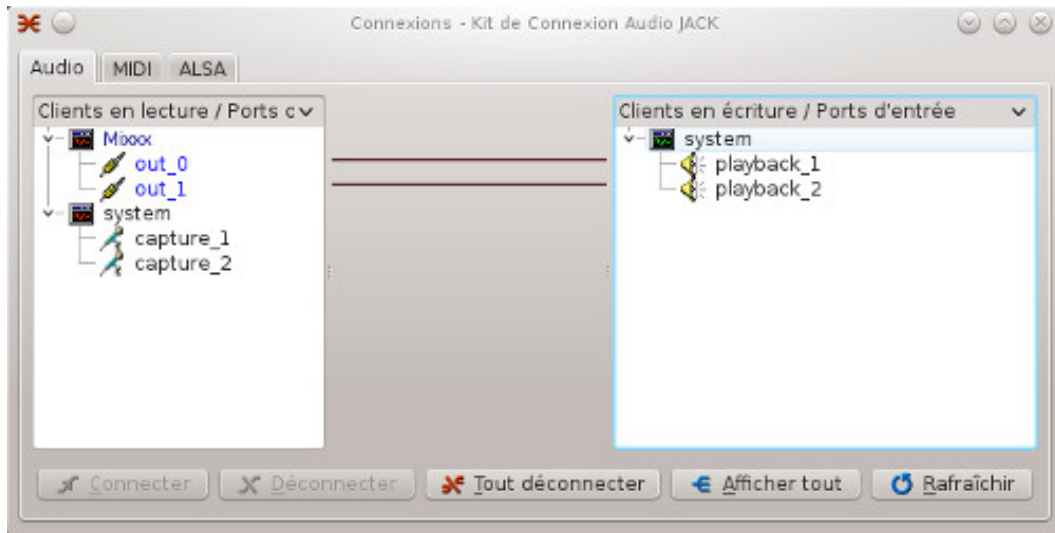
c'est déjà mieux, il faut aller maintenant dans les préférences et choisir comme contrôleur



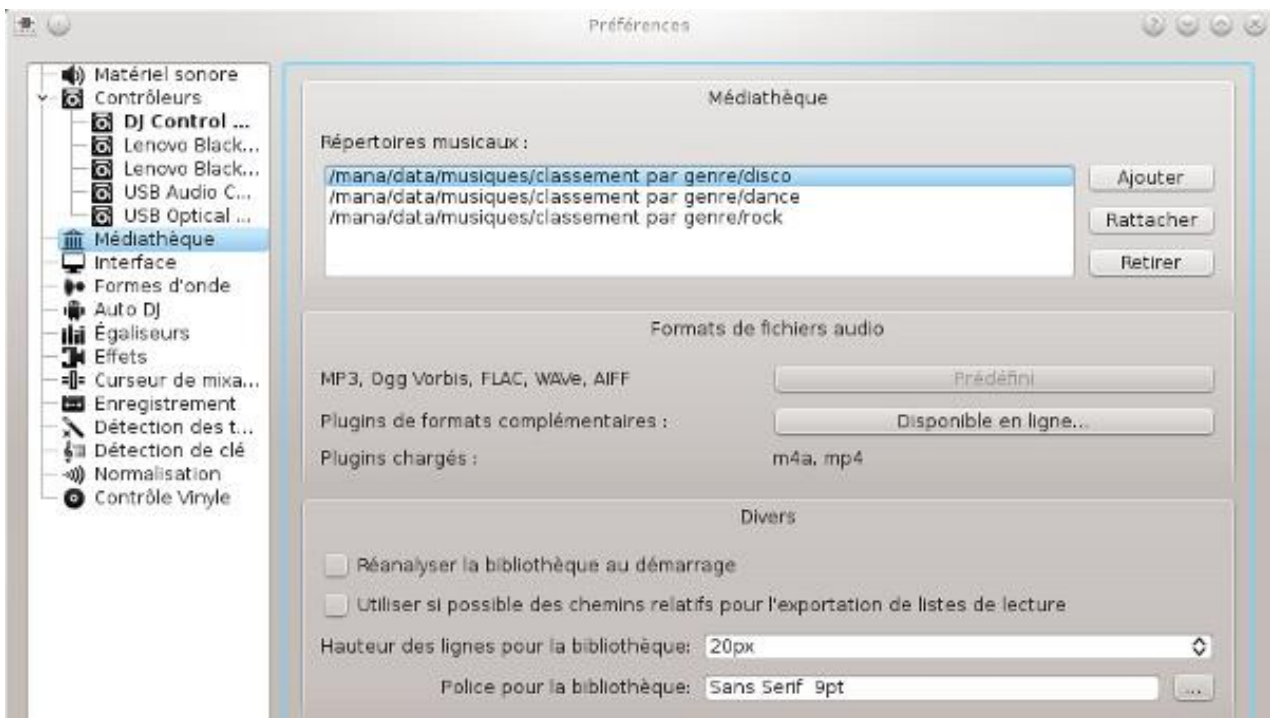
Au niveau de **Carte son** voilà mes préférences avec l'utilisation de **jackd**, en toute logique il faudrait pouvoir disposer de canaux 3 et 4 pour pouvoir brancher la sortie casque, mais pour cela il faut disposer d'une deuxième carte son ou d'une carte avec plusieurs sorties.



les connexions au niveau **qjackctl**



on indique ensuite l'endroit où se trouve la médiathèque musicale



Cela prendra un certain temps, le temps qu'il analyse la base de données musicale. Relancer **mixxx** et voilà ce qu'on devrait voir dans les traces

Debug [Main]: SoundManager::setupDevices()

Debug [Main]: SoundDevicePortAudio::open() "2, HDA ATI HDMI: 0 (hw:1,3)"

Debug [Main]: framesPerBuffer: 1024

Debug [Main]: Requested sample rate: 44100 Hz, latency: 23.22 ms

Debug [Main]: Output channels: 2 | Input channels: 0

Debug [Main]: Opening stream with id 2
Debug [Main]: Opened PortAudio stream successfully... starting
Debug [Main]: Dynamically loaded PortAudio library
Debug [Main]: PortAudio: Started stream successfully
Debug [Main]: Actual sample rate: 44100 Hz, latency: 23.22 ms
Debug [Main]: SoundDeviceNetwork::open() "Network stream"
Debug [Main]: framesPerBuffer: 1024
Debug [Main]: Requested sample rate: 44100 Hz, latency: 23.22 ms
Debug [Main]: Using "HDA ATI HDMI: 0 (hw:1,3)" as output sound device clock reference
Debug [Main]: 2 output sound devices opened
Debug [Main]: 0 input sound devices opened

voilà ce que ça donne



Le fonctionnement n'est pas sorcier, vous créez votre playlist à partir des menus en bas à gauche, puis vous chargez un morceau sur chacune des platines avec un simple drag and drop ensuite vous pilotez tout de la table Hercules, le mode d'emploi est strictement identique à celui fourni par la table.

7 Installation des plugins VST

VST est un protocole largement répandu pour les plugins audio, les plugins VST fournissent des fonctions de conversion de données audio et MIDI en temps réel. On distingue 3 sortes de plugins VST

- les plugins VST d'instrument qui simulent des instruments de musique qui peuvent être pilotés par

MIDI.

- les plugins **VST** d'effets audio, pour modifier les caractéristiques d'un son (reverb, distorsion, etc.)
- les plugins **VST** d'effets MIDI qui s'appliquent sur les commandes MIDI avant envoi vers l'instrument virtuel ou réel.

VST a été mis au point par la société [Steinberg](#), beaucoup de bibliothèques de plugins **VST** ne sont donc pas libres voire payantes, je présente donc deux bibliothèques libres et compatibles.

7.1 Computer Music Toolkit

CMT fournit des fonctions de traitement divers et variés sur l'audio, elle s'interface avec **LADSPA**. Le site officiel est <http://www.ladspa.org/download/> on y récupèrera les sources qu'on décompresse en tapant

```
tar xvfz cmt_src.tgz
```

cela donne le répertoire **cmt**, on se déplace dans le sous répertoire **src** et on modifie la ligne suivante du fichier **makefile**

```
INSTALL_PLUGINS_DIR = /usr/lib64/ladspa/
```

on tape ensuite

```
make
```

puis en tant que root on tape

```
make install
```

7.2 SWH

SWH est une suite d'outils de traitement audio compatibles avec **LV2**. Le site officiel est <http://plugin.org.uk/lv2/> on y récupèrera les sources dans lequel on tape

```
tar xvfz swh-lv2-1.0.15.tar.gz
```

cela donne le répertoire **swh-lv2-1.0.15** dans lequel on tape

```
make
```

puis en tant que root

```
make install
```

7.3 Tom's Audio Processing LADSPA plugins

Comme son nom l'indique Tom's Audio Processins s'interface avec LADPSA, on le récupèrera à cette adresse <http://sourceforge.net/projects/tap-plugins/files/latest/download> on récupèrera les sources qu'on décompresse en tapant

```
tar xvfz tap-plugins-0.7.3.tar.gz
```

cela donne le répertoire **tap-plugins-0.7.3** dans lequel on modifiera le fichier **Makefile** avec les lignes suivantes

```
INSTALL_PLUGINS_DIR = /usr/lib64/ladspa/  
INSTALL_LRDF_DIR    = /usr/share/ladspa/rdf/
```

on tape ensuite **make** puis en tant que root **make install**

8 MIDI

8.1 Présentation MIDI

MIDI est l'acronyme de Musical Instrument Digital Interface, ce n'est pas un codec audio comme le mp3 mais un protocole de communication entre des instruments de musique et/ou des ordinateurs. On trouvera davantage de détail sur [la page de wikipedia](#). Sans rentrer dans une lourde théorie, on peut utiliser un clavier/piano compatible midi (par prises midi ou usb) à partir d'un logiciel de MAO, on peut enregistrer sur le logiciel les commandes midi qui sont jouées sur l'instrument et inversement l'instrument peut restituer les commandes midi issues du logiciel.

Pour pouvoir entendre ce qui est joué, ça peut se faire via l'instrument directement ou via un synthétiseur MIDI comme on va le voir plus loin.

8.2 SoundFont

Les soundfonts sont un format de fichier particulier permettant de stocker des sons d'instruments ainsi que des informations sur le comportement du son. Elles sont utilisés pour les logiciels de MAO comme banque de sons. Les soundfonts servent notamment à restituer des fichiers MIDI, sans soundfonts vous ne pourrez pas écouter du MIDI sur les enceintes de votre PC.

Voilà quelques sites où on peut récupérer des soundfonts.

<http://musescore.org/en/handbook/soundfont>

<http://www.personalcopy.com/linuxfiles.htm>

8.3 FluidSynth

FluidSynth est un synthétiseur MIDI, le site officiel est <http://sourceforge.net/apps/trac/fluidsynth/> on y récupère les sources qu'on décompresse en tapant

```
tar xvfj fluidsynth-1.1.6.tar.bz2
```

cela donne le répertoire **fluidsynth-1.1.6** dans lequel on tape

```
./configure --enable-ladspa  
make
```

puis en tant que root

```
make install
```

8.4 Qsynth

Qsynth est une interface graphique à **FluidSynth**. Le site officiel est <http://qsynth.sourceforge.net/qsynth-index.html> on y récupère les sources qu'on décompresse en tapant

```
tar xvfz qsynth-0.4.3.tar.gz
```

cela donne le répertoire **qsynth-0.4.3** dans lequel on tape

```
./configure
```

si vous n'avez pas basculé **KF5** et **Qt5** rajoutez l'option **--enable-qt4**

```
make
```

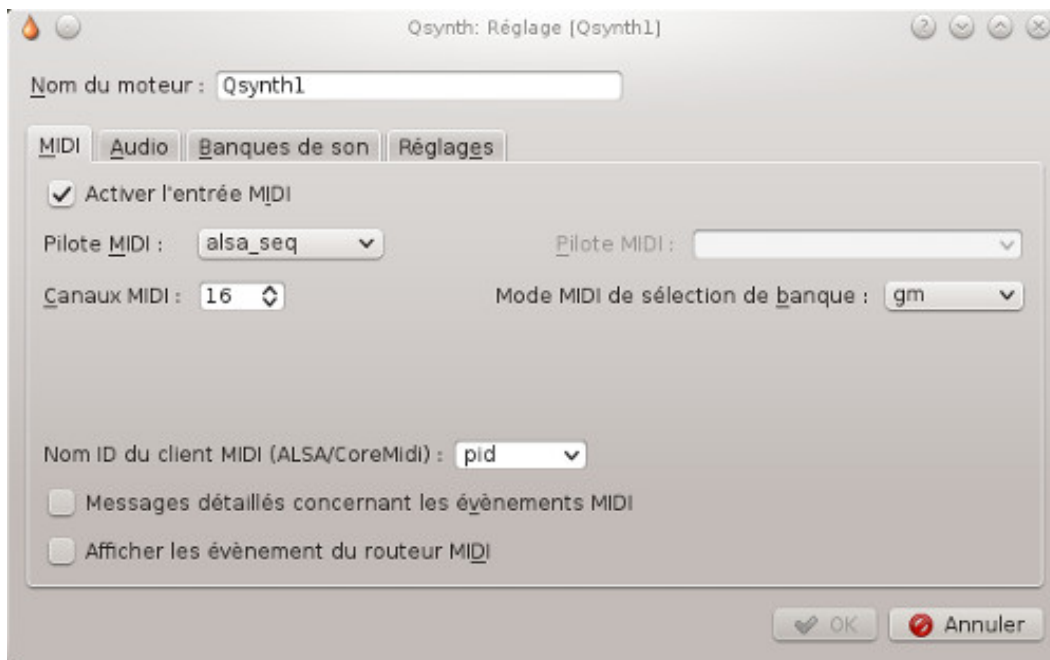
puis en tant que root

```
make install
```

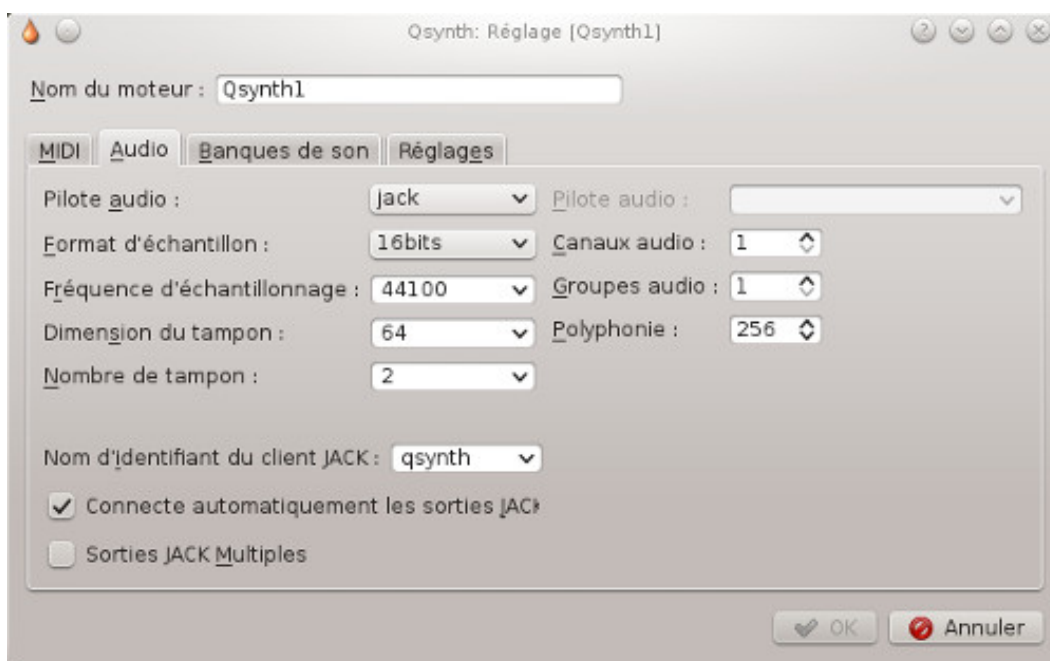
maintenant si on souhaite se passer de l'instrument MIDI et écouter la piste MIDI sur les hauts parleurs, il faudra lancer **jackd** avec **qjackctl** et configurer **Qsynth** on clique sur **Configuration**



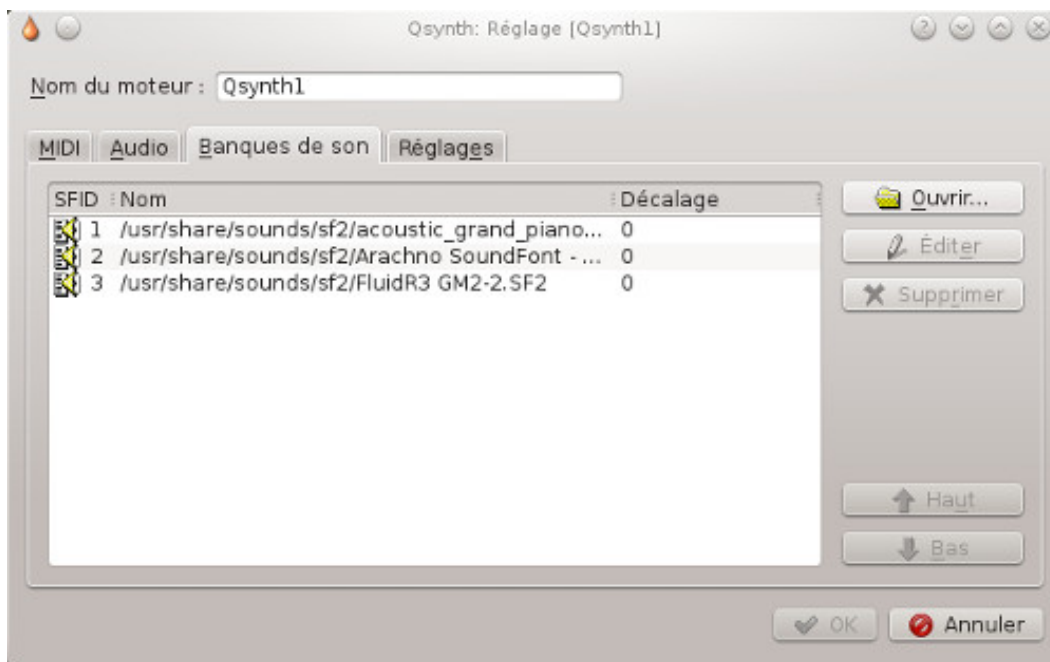
au niveau de l'onglet **MIDI** voici la configuration



l'onglet **Audio** avec **jackd**



au niveau de l'onglet **Banques de son** on charge les soundfonts récupérés préalablement.



c'est cette configuration type qui servira pour l'utilisation des softs.

8.5 Instruments MIDI

Je dispose d'un piano avec une interface MIDI via USB, il est automatiquement reconnu par le système voilà la trace avec **journalctl -f**

```
janv. 10 16:01:43 fakarava.kervao.fr kernel: usb 2-1.1.1.1: new full-speed USB device number 7 using ehci-pci
janv. 10 16:01:43 fakarava.kervao.fr kernel: usb 2-1.1.1.1: New USB device found, idVendor=0a67, idProduct=1011
janv. 10 16:01:43 fakarava.kervao.fr kernel: usb 2-1.1.1.1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
janv. 10 16:01:43 fakarava.kervao.fr kernel: usb 2-1.1.1.1: Product: Medeli e-Drum
janv. 10 16:01:43 fakarava.kervao.fr kernel: usb 2-1.1.1.1: Manufacturer: Medeli EI????
janv. 10 16:01:43 fakarava.kervao.fr mtp-probe[9840]: checking bus 2, device 7:
"/sys/devices/pci0000:00/0000:00:1d.0/usb2/2-1/2-1.1/2-1.1.1/2-1.1.1.1"
janv. 10 16:01:43 fakarava.kervao.fr mtp-probe[9840]: bus: 2, device: 7 was not an MTP device
```

Sur les logiciels de MAO comme **rosegarden**, il n'y a rien à faire, il est automatiquement reconnu.

9 Configuration système

Il va falloir modifier 2, 3 bricoles sur le PC pour pouvoir améliorer les performances de traitement en temps réel. Par exemple si au lancement d'**Ardour**, vous avez le message suivant

AVERTISSEMENT : Votre système possède une limite supérieure pour la quantité maximale de mémoire verrouillée.

Ceci peut empêcher Ardour d'accéder à de la mémoire avant que la mémoire disponible sur votre système ne soit épuisée.

Vous pouvez voir la limite actuelle avec la commande « ulimit -l » et celle-ci est habituellement contrôlée par le fichier /etc/security/limits.conf.

Il faudra éditer le fichier **/etc/security/limits.conf** et rajouter à la fin (ou modifier)

```
# Real Time support for audio group
```

```
@audio - rtprio 90
```

```
@audio - memlock unlimited
```

vos utilisateurs doivent être impérativement également dans le groupe **audio**. Maintenant on rajoute à la fin du fichier **/etc/rc.d/rc.local** les lignes suivantes

```
echo 2048 >/sys/class/rtc/rtc0/max_user_freq
```

```
echo 2048 >/proc/sys/dev/hpet/max-user-freq
```

```
echo 0 > /proc/sys/vm/swappiness
```

Maintenant sous **/lib/udev/rules.d** j'ai créé le fichier **40-rtc-permissions.rules** qui contient

```
KERNEL=="rtc0", GROUP="audio"
```

j'ai créé également le fichier **40-hpet-permissions.rules** qui contient

```
KERNEL=="hpet", GROUP="audio"
```

Maintenant j'ai édité le fichier **/etc/sysconfig/pulseaudio** et j'ai mis

```
PULSE_SERVER_TYPE=none
```

Pour ma configuration basée sur une [authentification avec LDAP](#), le fichier **limits.conf** n'était pas pris en compte j'ai du modifier ainsi le fichier **/etc/pam.d/system-auth**

```
session required pam_limits.so
```

```
session required pam_tcb.so
```

```
session sufficient pam_ldap.so
```

```
session optional pam_keyinit.so revoke
```

```
session [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
```

```
-session optional pam_systemd.so
```

pour mémoire, voilà ce que me donne le branchement de l'UCA222

```

Dec 29 14:18:09 mana kernel: [ 6455.501032] usb 1-6: new full-speed USB device number 3
using ohci_hcd
Dec 29 14:18:09 mana kernel: [ 6455.691423] usb 1-6: New USB device found,
idVendor=08bb, idProduct=2902
Dec 29 14:18:09 mana kernel: [ 6455.691428] usb 1-6: New USB device strings: Mfr=1,
Product=2, SerialNumber=0
Dec 29 14:18:09 mana kernel: [ 6455.691430] usb 1-6: Product: USB Audio CODEC
Dec 29 14:18:09 mana kernel: [ 6455.691432] usb 1-6: Manufacturer: Burr-Brown from
TI
Dec 29 14:18:10 mana kernel: [ 6455.703519] input: Burr-Brown from TI          USB Audio
CODEC as /devices/pci0000:00/0000:00:02.0/usb1/1-6/1-6:1.3/input/input9
Dec 29 14:18:10 mana kernel: [ 6455.703677] generic-usb 0003:08BB:2902.0002:
input,hidraw1: USB HID v1.00 Device [Burr-Brown from TI          USB Audio CODEC ]
on usb-0000:00:02.0-6/input3
Dec 29 14:18:10 mana mtp-probe: checking bus 1, device 3:
"/sys/devices/pci0000:00/0000:00:02.0/usb1/1-6"
Dec 29 14:18:10 mana mtp-probe: bus: 1, device: 3 was not an MTP device
Dec 29 14:18:10 mana kernel: [ 6455.930434] 3:0: cannot get min/max values for control 2 (id
3)
Dec 29 14:18:10 mana kernel: [ 6455.930821] usbcore: registered new interface driver snd-
usb-audio

```

et voilà le résultat de la commande

```
more /proc/asound/cards
```

```

0 [NVidia      ]: HDA-Intel - HDA NVidia
                  HDA NVidia at 0xdbef4000 irq 22
1 [CODEC       ]: USB-Audio - USB Audio CODEC
                  Burr-Brown from TI USB Audio CODEC at usb-0000:00:02.0-6, full speed

```

Voilà ce que ça donne en branchant le Behringer Xenyx 302

```

Jan  4 11:25:47 mana kernel: [ 349.944028] usb 2-6: new full-speed USB device number 4
using ohci_hcd
Jan  4 11:25:47 mana kernel: [ 350.134090] usb 2-6: New USB device found, idVendor=08bb,
idProduct=2902
Jan  4 11:25:47 mana kernel: [ 350.134094] usb 2-6: New USB device strings: Mfr=1,
Product=2, SerialNumber=0
Jan  4 11:25:47 mana kernel: [ 350.134097] usb 2-6: Product: USB Audio CODEC
Jan  4 11:25:47 mana kernel: [ 350.134099] usb 2-6: Manufacturer: Burr-Brown from
TI
Jan  4 11:25:47 mana kernel: [ 350.221095] 3:0: cannot get min/max values for control 2 (id
3)

```

```
Jan 4 11:25:47 mana kernel: [ 350.230157] input: Burr-Brown from TI      USB Audio
CODEC as /devices/pci0000:00/0000:00:02.0/usb2/2-6/2-6:1.3/input/input10
Jan 4 11:25:47 mana kernel: [ 350.230360] generic-usb 0003:08BB:2902.0003:
input,hidraw2: USB HID v1.00 Device [Burr-Brown from TI      USB Audio CODEC ]
on usb-0000:00:02.0-6/input3
Jan 4 11:25:47 mana mtp-probe: checking bus 2, device 4:
"/sys/devices/pci0000:00/0000:00:02.0/usb2/2-6"
Jan 4 11:25:47 mana mtp-probe: bus: 2, device: 4 was not an MTP device
```

10 Outils de MAO

10.1 *MusE*

MusE est un logiciel complet de MAO, le site officiel est <http://muse-sequencer.org/> on y récupérera les sources qu'on décompresse en tapant

```
tar xvfz muse-2.2.1.tar.gz
```

cela donne le répertoire **muse-2.2.1** dans lequel on tape successivement

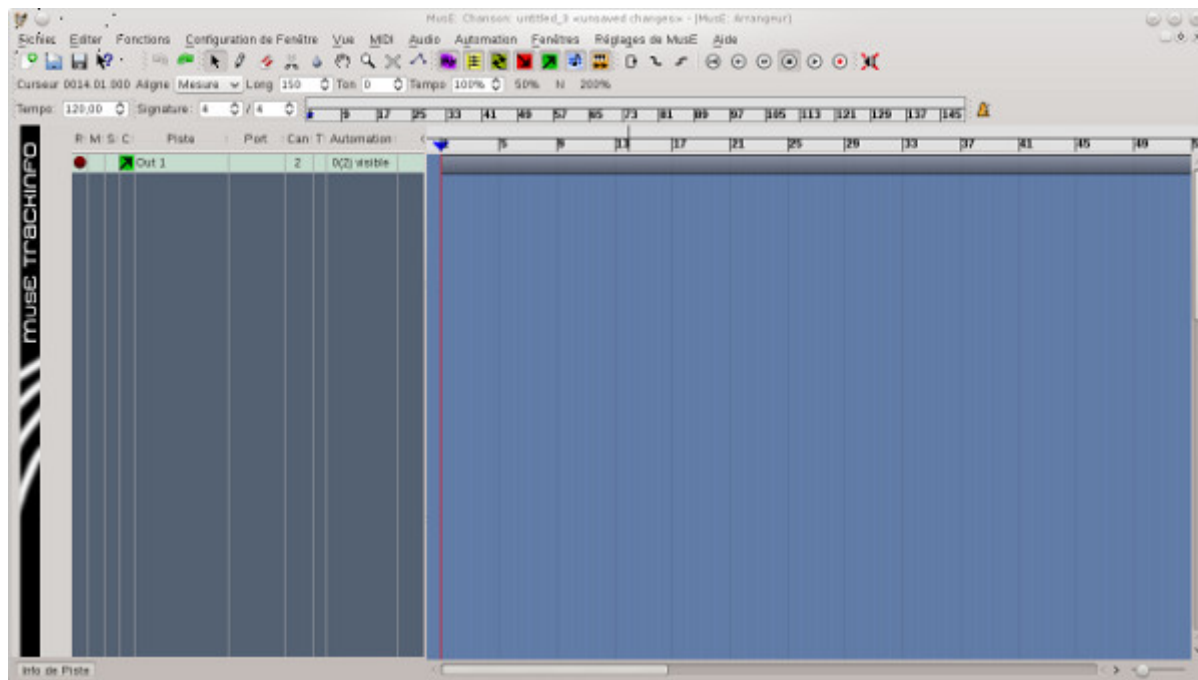
```
mkdir build
cd build
```

on devra installer préalablement le package **lib64lo-devel** puis on tape

```
cmake ..
make
```

puis en tant que root **make install**

On le lance en tapant **muse2** voilà la fenêtre principale



10.2 Linux MultiMedia Studio

LMMS est un outil très puissant pour faire ses créations audio, on peut y interfacer des périphériques MIDI, cela dit je n'ai pas trouvé la possibilité d'interfacer un instrument réel pour enregistrement. Pour l'installation on installera préalablement les packages **lib64ftk-devel** et **lib64fftw-devel**. Sur le site officiel <http://lmms.sourceforge.net/download.php> on récupèrera les sources qu'on décompresse en tapant

```
tar xvfj lmms-1.1.3.tar.bz2
```

cela donne le répertoire **lmms-1.1.3** dans lequel on tape successivement

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
make
```

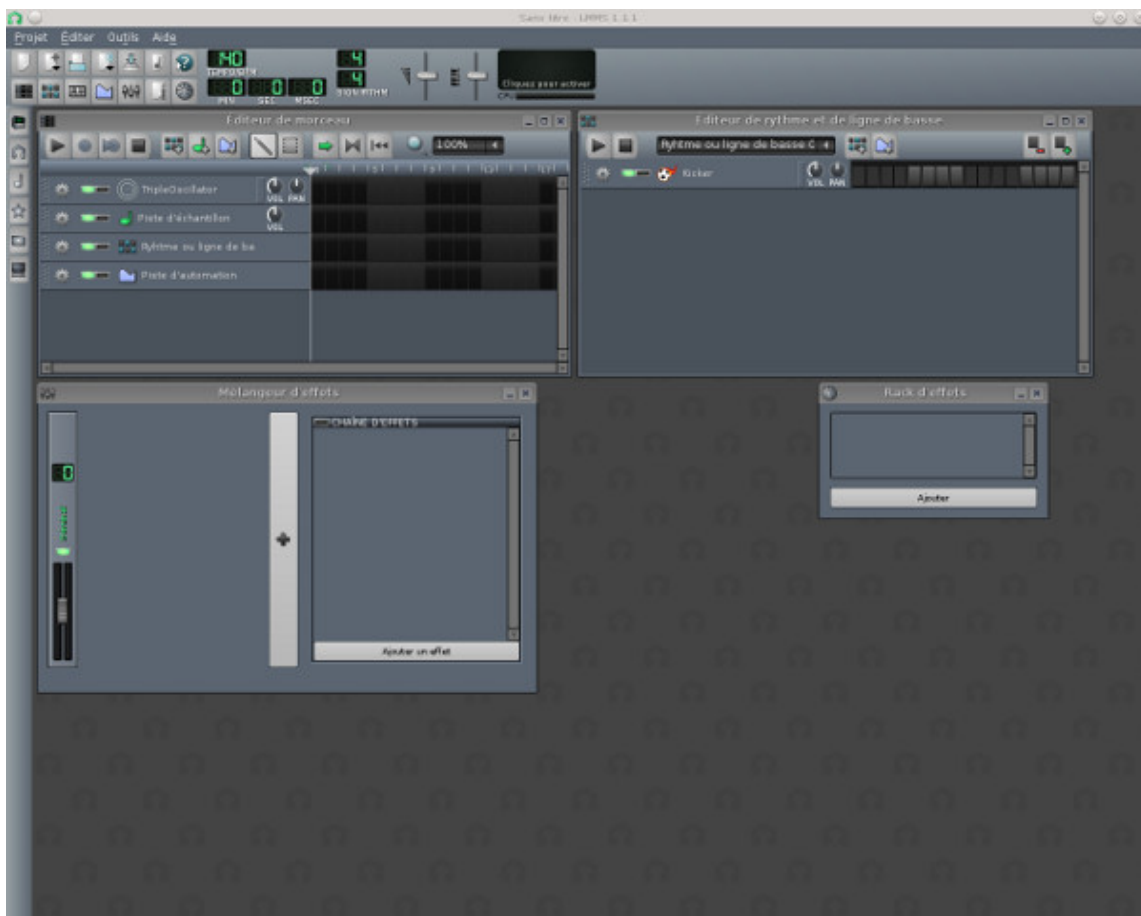
puis en tant que root

```
make install
```

Au premier lancement de **lmms** la fenêtre configuration apparaît. J'ai choisi l'interface audio **JACK**.



Voilà à quoi ressemble l'interface générale au démarrage



11 ardour

Ardour n'est pas vraiment simple d'utilisation, les commandes ne sont franchement pas intuitives par rapport à [rosegarden](#) par exemple. On peut brancher à peu près tout et n'importe quoi (voix, instruments réels, périphériques MIDI, ...), on peut rajouter des sons extérieurs (format .wav), il dispose de nombreux plugins.

Préalablement j'ai dû installer les packages suivants **scons**, **lib64cppunit-devel**, **lib64gnomecanvas2-devel**, **lib64lo-devel**, **lib64lrdp-devel**, **lib64glibmm2.4-devel**, **lib64archive-devel**, **lib64gtkmm3.0-devel** et **lib64gnomecanvasmm2.6-devel**

Le site officiel est <http://www.ardour.org/> on y récupère l'archive qu'on décompresse en tapant

```
tar xvfj Ardour-5.5.0.tar.bz2
```

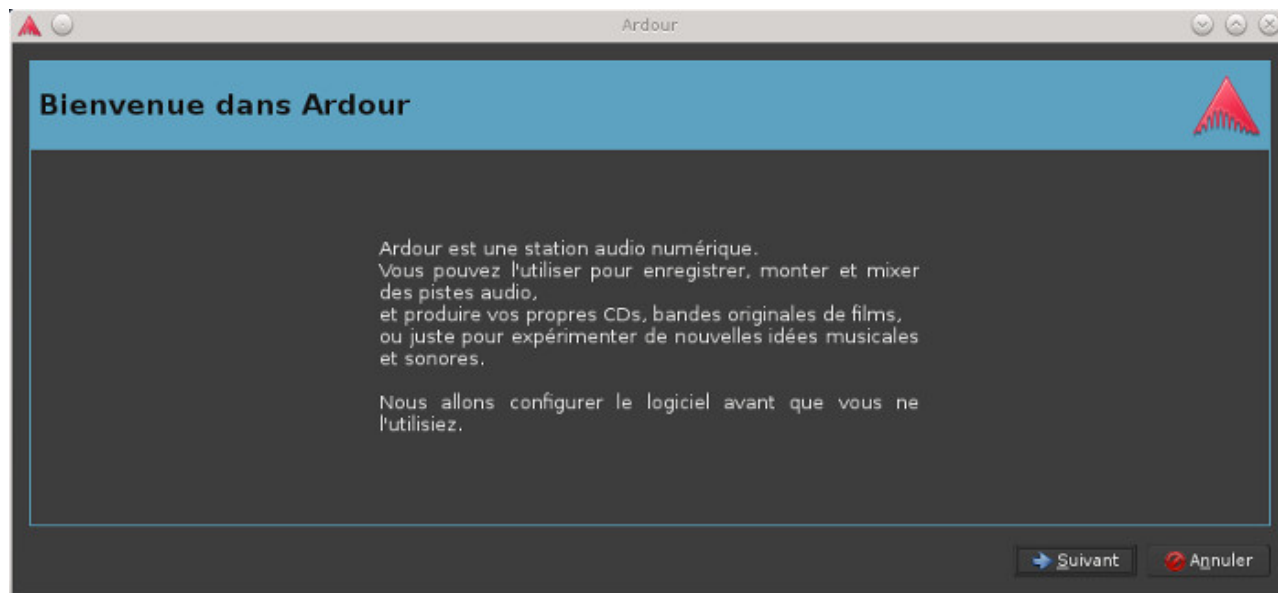
cela donne le répertoire **Ardour-5.5.0** dans lequel on tape

```
./waf configure --lv2
```

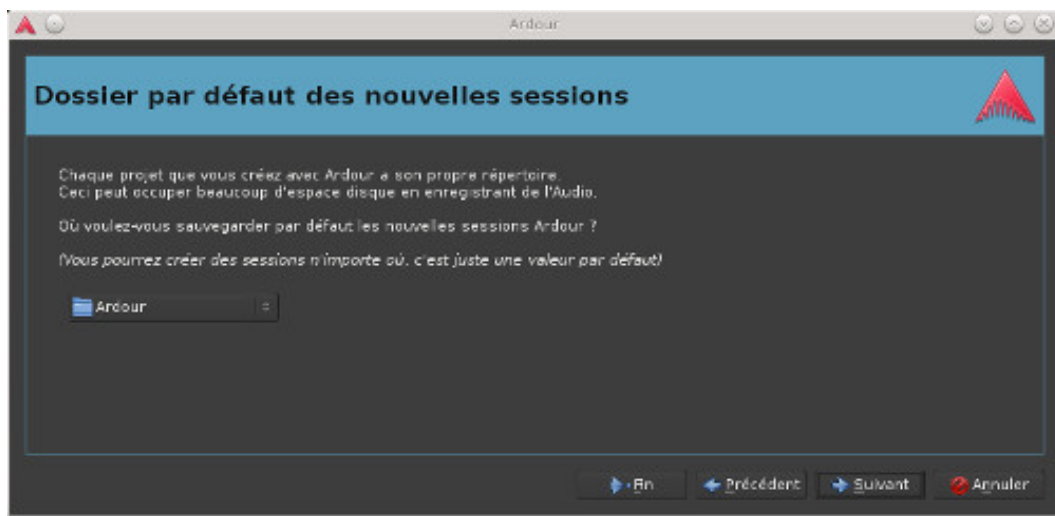
on tape ensuite **./waf** puis en tant que root

```
./waf install
```

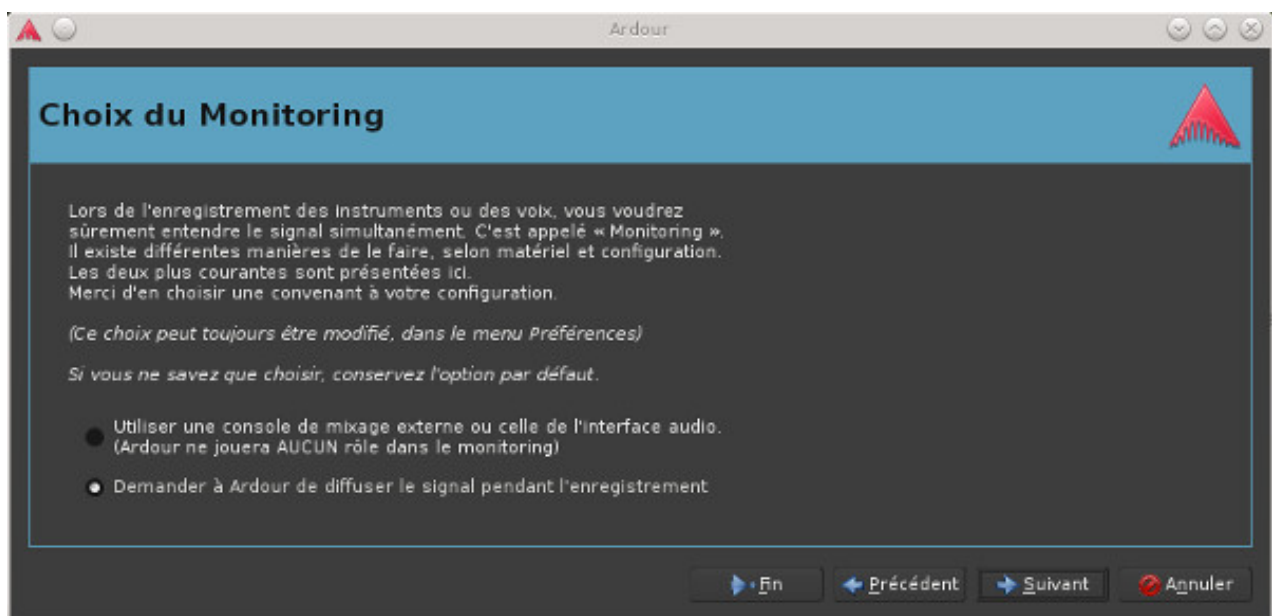
lors du premier lancement, on va passer par une phase de configuration



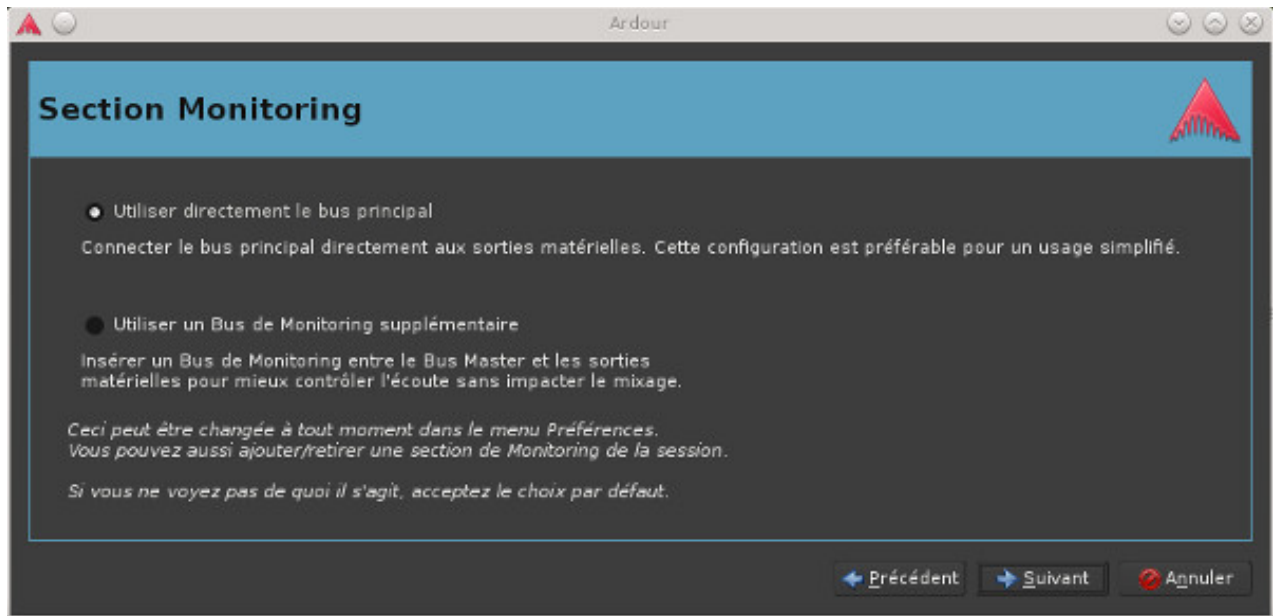
il va demander un répertoire de travail, j'ai créé pour cela préalablement un répertoire **Ardour** dans ma homedirectory



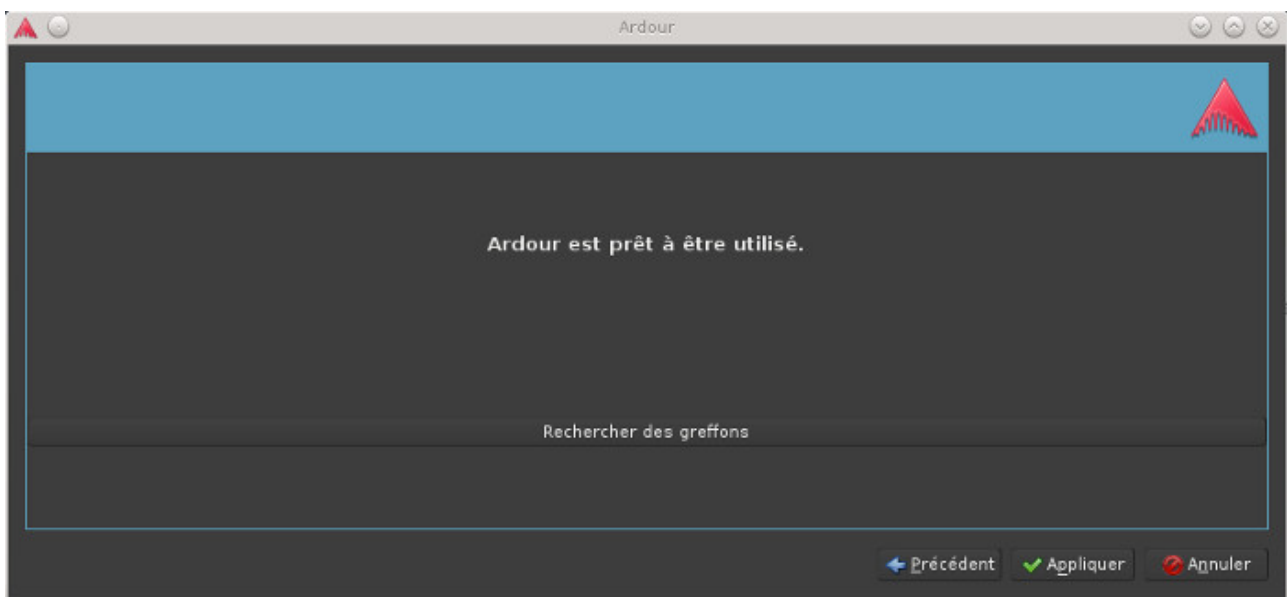
ensuite il faut indiquer comment on veut écouter les compositions, via **ardour** ou un outil externe, j'ai choisi de passer par **ardour**.



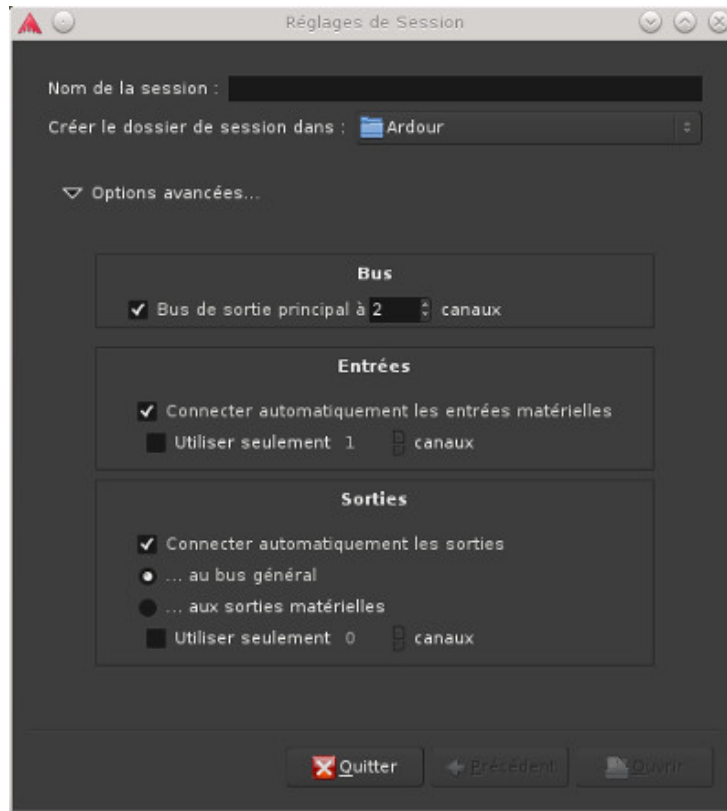
Pour la fenêtre **Section Monitoring** j'ai encore laissé le choix par défaut.



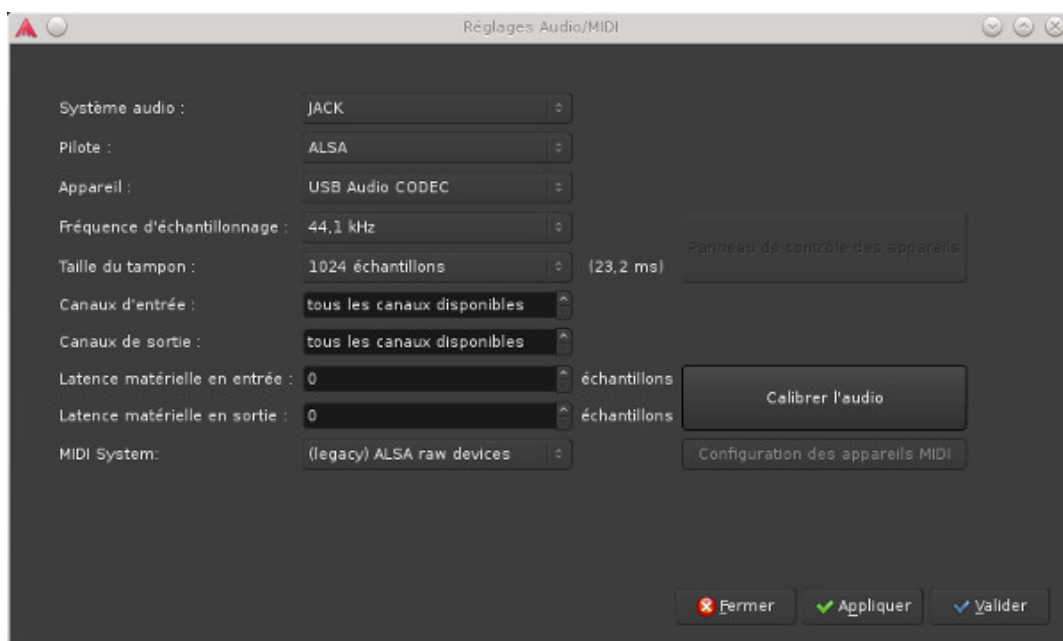
A la fenêtre suivante, il faudra cliquer sur **Rechercher des greffons**



Ardour est maintenant prêt à être utilisé. Il faut créer ensuite un nom de projet



A la création d'une nouvelle session d'enregistrement, il faudra configurer préalablement l'audio et le midi. Vu que j'ai 2 interfaces audio, je veux utiliser mon behringer xenyx 302 en entrée et ma carte son interne en sortie, il se trouve que ça ne fait pas bon ménage et je n'arrive pas avec **Ardour** à faire la distinction entre les deux, c'est soit l'une soit l'autre, alors du coup je dois lancer **qjackctl** en avance de phase et bien indiquer mon xenyx en entrée et ma carte son interne en sortie.



12 Qtractor

Qtractor est également un logiciel de MAO avec lequel on peut enregistrer des instruments réels de musique. Il est bardé également d'un tas de plugins audio. Le site officiel est <http://qtractor.sourceforge.net/qtractor-index.html> on y récupère les sources qu'on décompresse en tapant

```
tar xvfz qtractor-0.6.5.tar.gz
```

cela donne le répertoire **qtractor-0.6.5** dans lequel on tape successivement

```
./configure
```

Voilà le résultat

Qtractor 0.6.5

```
Build target . . . . .: release
```

```
JACK Audio Connection Kit support . . . . .: yes
```

```
ALSA MIDI Sequencer support . . . . .: yes
```

```
General audio file support (libsndfile) . . . . .: yes
```

```
Ogg Vorbis audio file support (libvorbis) . . . . .: yes
```

```
MPEG-1 Audio Layer 3 file support (libmad) . . . . .: yes
```

```
Sample-rate conversion support (libsamplerate) . . .: yes
```

```
Pitch-shifting support (librubberband) . . . . .: yes
```

```
OSC service support (liblo) . . . . .: yes
```

```
Archive/Zip file support (zlib) . . . . .: yes
```

```
IEEE 32bit float optimizations . . . . .: yes
```

```
SSE optimization support (x86) . . . . .: yes
```

```
LADSPA plug-in support . . . . .: yes
```

```
DSSI plug-in support . . . . .: yes
```

```
VST plug-in support . . . . .: yes
```

```
LV2 plug-in support (liblilv) . . . . .: yes
```

```
LV2 plug-in UI support (libsuil) . . . . .: yes
```

```
LV2 plug-in External UI support . . . . .: yes
```

```
LV2 plug-in MIDI/Event support . . . . .: yes
```

```
LV2 plug-in MIDI/Atom support . . . . .: yes
```

```
LV2 plug-in Worker/Schedule support . . . . .: yes
```

```
LV2 plug-in State support . . . . .: yes
```

```
LV2 plug-in State Files support . . . . .: yes
```

```
LV2 plug-in Programs support . . . . .: yes
```

```
LV2 plug-in Presets support . . . . .: yes
```

```
LV2 plug-in Time support . . . . .: yes
```

```
LV2 plug-in Time/position support . . . . .: yes
```

LV2 plug-in Options support: yes
LV2 plug-in Buf-size support: yes
LV2 plug-in UI Idle interface support: yes
LV2 plug-in UI Show interface support: yes

JACK Session support: yes
JACK Latency support: yes
JACK Metadata support: yes

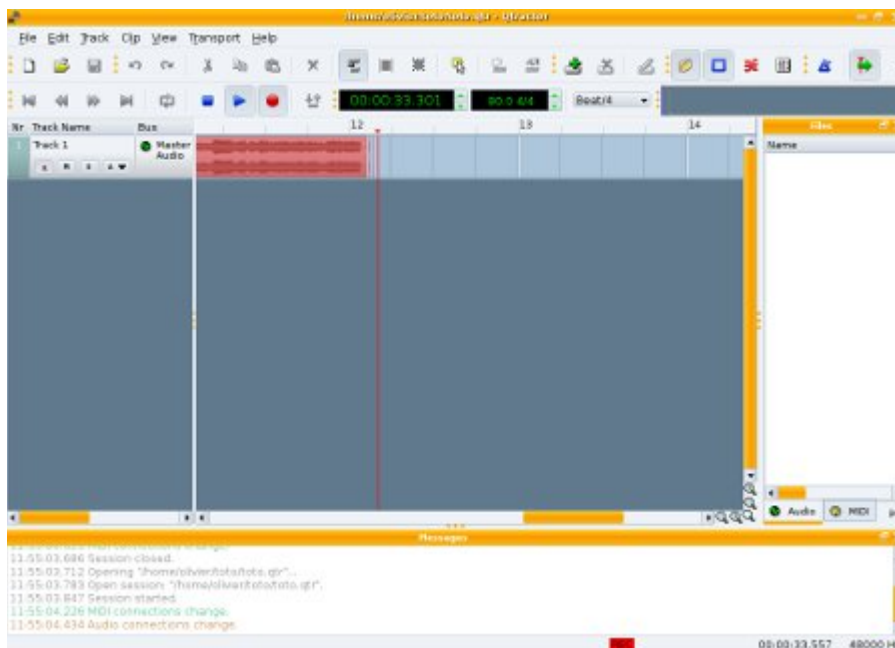
Non Session Management (NSM) support: yes

X11 Unique/Single instance: no
VeSTige header support: yes
Gradient eye-candy: yes
Debugger stack-trace (gdb): no

Install prefix: /usr/local

on tape maintenant **make** puis en tant que root

make install



Pour aller plus loin un tutorial <http://www.slackermidia.info/qtractor/index.html> (en anglais désolé)

13 Rosegarden

Rosegarden est également extrêmement puissant, il permet également l'enregistrement d'instruments réels, je n'ai pas trouvé cependant comment importer des fichiers audio en .wav. J'ai du installer préalablement le package **makedepend**. Le site officiel est

<http://www.rosegardenmusic.com/> on y récupèrera les sources qu'on décompressera en tapant

```
tar xvfj rosegarden-16.06.tar.bz2
```

Cela donne le répertoire **rosegarden-16.06** dans lequel on tape

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
make
```

puis en tant que root

```
make install
```

Pour les plugins **LADPSA** il va les chercher par défaut sous **/usr/lib64/ladspa/**. Si vos plugins se trouvent ailleurs par exemple sous **/usr/local/lib/ladspa** vous pouvez modifier la variable d'environnement **LADSPA_PATH** de cette manière.

```
export LADSPA_PATH=$LADSPA_PATH:/usr/local/lib/ladspa/
```

par contre il ne supporte toujours pas les plugins **LV2** ce qui est quand même son gros handicap.

Voilà la fenêtre générale de **rosegarden**

