

WERKEN MET QJACKCTL OP GNU/LINUX

(HANDLEIDING VERSIE 4.0)

ROSEA GRAMMOSTOLA

ROSEA.GRAMMOSTOLA@GMAIL.COM



Contents

1	Introductie	3
2	JACK	4
3	QjackCTL	5
3.1	Setup	5
3.1.1	Balans latency en xruns	5
3.1.2	Instellingen	6
3.1.3	Gebruik	8
3.1.4	Start	8
3.1.5	Messages	9
3.2	Connect	9
3.2.1	Audio	9
3.2.2	Voorbeeld	10
3.2.3	MIDI ALSA	11
3.2.4	MIDI JACK	11
3.3	De Patchbay	13
3.3.1	Wat is de Patchbay precies?	13
3.3.2	Een voorbeeld van een patchbay	13
3.3.3	Het maken van de Patchbay	13
3.3.4	Het toevoegen van Sockets en Plugs	15
3.3.5	Verbinding maken	17
3.3.6	Opslaan en activeren	17
3.4	Jack synchronisatie	19
3.4.1	JACK Transport Control System	19
3.4.2	Voorbeeld: Hoe Hydrogen synchroniseren met Ardour	20
3.4.3	Vele mogelijkheden	21
4	Achtergrond informatie	22
4.0.4	Met dank aan	22

4.0.5	Referenties	22
-------	-----------------------	----

Chapter 1

Introductie

Als je muziek wilt maken op Linux dan kun je dat in principe doen op elke ‘normale’ Linux distributie zonder verdere speciale configuratie. Met programma’s als Audacity, Hydrogen, LMMS, MuseScore en Mixxx bijvoorbeeld kun je datgene doen waar het programma voor gemaakt is: muziek maken!

Als je echter serieus muziek wil componeren en produceren op Linux en wilt werken met programma’s als Ardour en Rosegarden dan zul je kennis moeten maken met iemand die ik enkele jaren geleden heb leren kennen en inmiddels een goede vriend van me is geworden: JACK.

Chapter 2

JACK

JACK, of wel ‘JACK audio connection kit’, is een low-latency audio server voor Linux en Apple’s OS X. JACK maakt het mogelijk om verschillende programma’s te verbinden aan een audio device. De zogenaamde JACK clients (programmas met JACK ondersteuning zoals, Ardour, Rosegarden, Audacity en Hydrogen) kunnen dus op zichzelf draaien maar ook binnen de JACK server (e.g. als een soort ‘plugin’). JACK is ontworpen voor professionele audio productie en het ontwerp focust zich op twee centrale gebieden:

1. Low latency
2. Gelijktijdig afspelen (synchroniseren) van JACK clients.

(Low)Latency duidt aan hoeveel vertraging er optreedt tussen de audio input (het spelen van een toon op je MIDI-keyboard bijvoorbeeld) en het horen van de output, de gespeelde toon in dit geval. Uiteraard wil je die vertraging zo kort mogelijk hebben en daar kan JACK je bij helpen.

Een voorbeeld van de rol van JACK: Je sluit je gitaar en microfoon aan op je multiple-input geluidskaart. Je MIDI keyboard kun je aansluiten aan een USB poort voor MIDI opnames. De audio (van gitaar en microfoon) kan opgenomen worden in een DAW (bijv. Ardour), terwijl de MIDI opnames worden gemaakt in een MIDI sequencer (bijv. Qtractor of Rosegarden). Uiteraard wil je ook een mooie drumpartij bij de song hebben en dus gebruik je een drumcomputer (bijv. Hydrogen). De output van de verschillende programmas (het geluid van je gitaar en microfoon, de MIDI opname in Qtractor en de drums gespeeld in Hydrogen) kunnen gelijktijdig opgenomen worden als een stereo WAV file in een programma zoals Ardour. JACK zorgt in dit proces voor low latency (zo klein mogelijke vertraging), het verbinden van de verschillende en het synchroniseren (gelijktijdig afspelen) van de verschillende programma’s. JACK werkt in principe middels de ‘command line’ (terminal), maar gelukkig is er een goede GUI voor JACK: QjackCTL.

Chapter 3

QjackCTL

Qjackctl (ook wel ‘JACK control’) kun je starten via het menu of door qjackctl in je terminal te typen. Als het goed is zie je dan het volgende venster verschijnen.



Figure 3.1: QjackCTL

3.1 Setup

Om Qjackctl (we hebben het hier over versie: 0.3.3) te kunnen starten en om de maximale prestatie uit JACK te halen, is het belangrijk dat je de Setup goed ingesteld hebt. Dit kun je doen door op de knop Setup te klikken. Er verschijnt nu een venster zoals afgebeeld in figuur 3.2 op de volgende pagina.

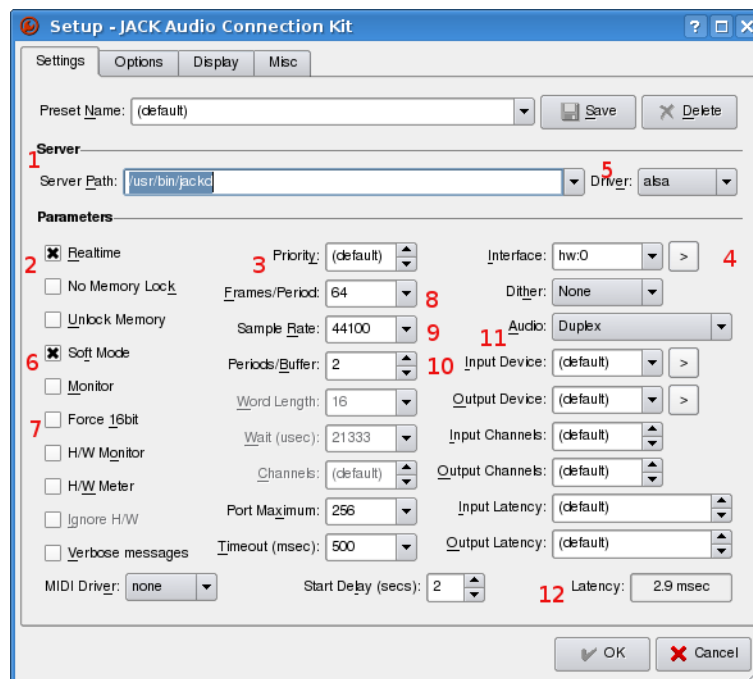
3.1.1 Balans lantency en xruns

Bij het instellen van de Setup voor QjackCTL is het belangrijk dat je een ding in de gaten houdt en dat is de balans tussen latency en de zogenaamde xruns. Het idee is om JACK op zo’n manier te laten werken dat de latency (vertraging) zo klein mogelijk is, zonder dat dit zorgt voor constante xruns. Xruns zijn als het ware kleine storingen die veroorzaakt worden doordat een bepaald proces JACK verstoort of dat JACK een constante stroom van audio data niet meer aan kan. Een xrun is hinderlijk omdat daardoor de opname verstoord wordt. Het is vrij normaal dat je enkele xruns krijgt als je een programma start of afsluit, maar tijdens de

opname kan het voor een klik of het wegvallen van geluid zorgen en dat wil je uiteraard niet. Vandaar dat er gesproken wordt over de balans tussen latency en xruns. Je wilt een zo laag mogelijke latency, maar ook een minimaal aantal xruns. (Voor een zo goed mogelijke werking van JACK en een zo laag mogelijke latency is het aan te raden om gebruik te maken van een Linux distributie welke speciaal geconfigureerd is voor multimedia productie zoals 64studio, Jacklab of Ubuntu Studio. Met name het gebruik van een zogenaamde RealTime kernel (rt-kernel) kan de (realtime) latency verminderen. Uiteraard kun je zelf ook een rt-kernel installeren en de distro die je gebruikt configureren voor het gebruik van JACK.)

3.1.2 Instellingen

Niet alle instellingen die je op afbeelding 3.2 ziet (vooral als het gaat om de Parameters section), zullen bij jou hetzelfde zijn. Dit is een afbeelding van de instellingen specifiek voor mijn hardware, dit kan bij jouw wat anders zijn als je bijvoorbeeld een andere geluidskaart hebt. Hieronder de punten die je eventueel moet instellen om goed te kunnen werken met JACK.



Figuur 3.2: Setup

1. Zorg er allereerst voor dat de Server Path naar jackd verwijst (/usr/bin/jackd). Tenzij je met jackdmp werkt, dan moet je naar jackdmp laten verwijzen. Als je niet weet wat jackdmp is, dan mag je deze laatste opmerking negeren).

2. Zorg dat de Realtime Mode aangevinkt is.
3. Zorg er ook voor dat de priority op 0 (default) gezet is. Dit zorgt ervoor dat alle audio processen prioriteit hebben zodat je de beste prestatie voor je productie krijgt.
4. De volgende stap is te zorgen dat de juiste geluidskaart ingesteld is. Dit kan onder settings : (parameters) : interface. Waarschijnlijk is (default) goed, maar er zit ook een knopje waar je een gevonden geluidskaarten op je systeem kunt aanvinken. Bij mij is dat default of hw:0. Normaal gesproken is het goed om degenen aan te vinken met de model naam van je kaart. (Als JACK niet wil starten dan zal je wat moeten experimenteren met de kaart keuze, maar dat is eventueel van latere zorg...).
5. Als je geluidskaart niet zichtbaar is, dan is de kaart wellicht niet compatibel met ALSA. Probeer dan een andere driver in de Server section van het Setup venster (OSS of pulseaudio(?)). De meeste kaarten zijn compatibel met ALSA en het is aan te raden om ALSA als driver te kiezen, tenzij je een Firewire interface hebt, dan zul je de Freebob driver moeten kiezen.
6. Je kunt Softmode aanvinken, dat betekent dat JACK niet stopt wanneer je Xruns krijgt, wel handig als je aan het opnemen bent dus.
7. Als je zeker weet dat je computer alleen werkt met een 16bit resolutie zorg dan dat Force 16bit aangevinkt is.
8. De instelling van Frames/Period, in de middelste kolom van het Setup venster, heeft de meeste invloed op de hoeveelheid latency (vertraging). Hoe hoger de Frames/ Period hoe hoger de latency. Je zult dus moeten kijken wat je geluidskaart en systeem aan kan. Ik heb Frames/ Period nu (Ik heb een M-audio audiophile 24/96) staan op 32 of 64, maar met mijn 'onboard' geluidskaart had ik het op 128 of 256 staan...
9. Sample Rate staat op 44100 (CD kwaliteit) maar sommige geluidskaarten prefereren 48000 (DAT kwaliteit).
10. Periods/Buffer moet zeer waarschijnlijk op 2 gezet worden, maar het kan zijn dat het met jouw geluidskaart net iets anders is.
11. Audio kun je het beste op Duplex zetten, mocht JACK niet starten dan kun je eens alleen Playback only proberen.
12. Rechts onderin het venster Setup : settings zie je een label met latency en de waarde die daarbij hoort.

Aan de andere instellingen kun je het beste (voorlopig) niet sleutelen.

3.1.3 Gebruik

We kunnen tot slot QjackCTL zo instellen dat het wat makkelijker in gebruik wordt. We moeten daarvoor twee opties aanvinken in het tabblad Misc onder Setup (zie figuur 3.3):

1. Enable system tray icon;
2. Start Jack audio server on application start up.



Figuur 3.3: Misc

3.1.4 Start

Laten we nu eens gaan kijken of we JACK kunnen starten met behulp van QjackCTL. We sluiten het Setup venster af (save, ok). Om JACK te starten moeten we, hoe verrassend, de start button aanklikken. Belangrijk bij het starten van JACK is dat andere geluid applicaties gesloten zijn. JACK is namelijk erg jaloers en duldt geen andere sound servers naast zich. Over het algemeen is het verstandig om zoveel mogelijk (software) processen op je systeem af te sluiten voor een goede JACK performance. Een website als YouTube bijvoorbeeld zorgt er voor dat JACK niet kan starten... Start JACK ook altijd VOOR je de JACK clients start! Als

alles goed is ingesteld, start je JACK en verschijnt in het schermpje ‘Started’ (zie figuur 3.4 op de pagina hierna).



Figuur 3.4: Start

3.1.5 Messages

Als het werkt, maar zeker als het niet werkt, kun je bij ‘Messages’ kijken of er iets nuttigs gemeld wordt. Bij foutmeldingen of veel xruns kun je wat veranderen aan je settings, bijvoorbeeld Frames/ Period omhoog doen en anders via een zoekmachine op het internet zoeken naar de foutmelding of op fora of mailinglists om assistentie vragen (Nederlandstalig Linux forum, Linux Musicians Forum of Linux Audio User Mailinglist.)

3.2 Connect

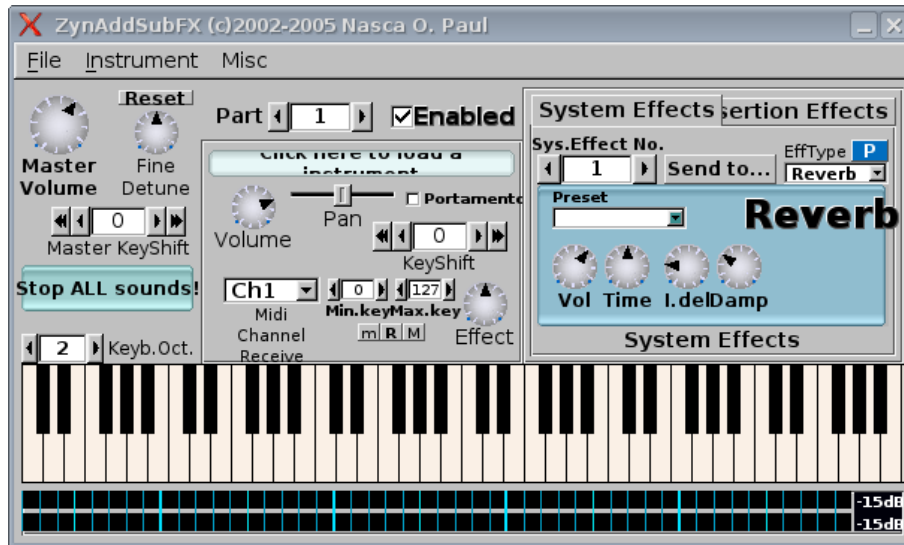
3.2.1 Audio

Als JACK draait dan kunnen we eens proberen om een zogenaamde JACK client te starten. Dit kunnen allerlei audio software zijn die ondersteuning voor JACK hebben. Een goed programma voor onze test is ZynAddSubFx (zorg dat het geïnstalleerd is). Start het programma op via het menu of door `zynaddsubfx` in je terminal te typen. ZynAddSubFx start nu op in de ‘Beginner Mode’ = met een toetsenbord (is dit niet het geval switch dan in ZynAddSubFx via Misc naar de ‘Beginner Mode’) en maakt het automatisch een connectie met JACK.

Om te kunnen zien of er inderdaad een connectie tot stand is gekomen moet je op de knop Connect klikken in QjackCTL. Als het goed is dan zie je het volgende venster verschijnen (zie figuur 3.6 op pagina 11).

Als er een verbinding is gemaakt en je hoort geluid via je speakers, koptelefoon of monitoren, dan kunnen we concluderen dat JACK werkt. Hieronder meer informatie over de JACK connections of verbindingen.

Zoals gezegd kun je met behulp van JACK allerlei verbindingen maken tussen de verschillen audio/ MIDI software/ hardware, de JACK clients. In het Audio tabblad worden



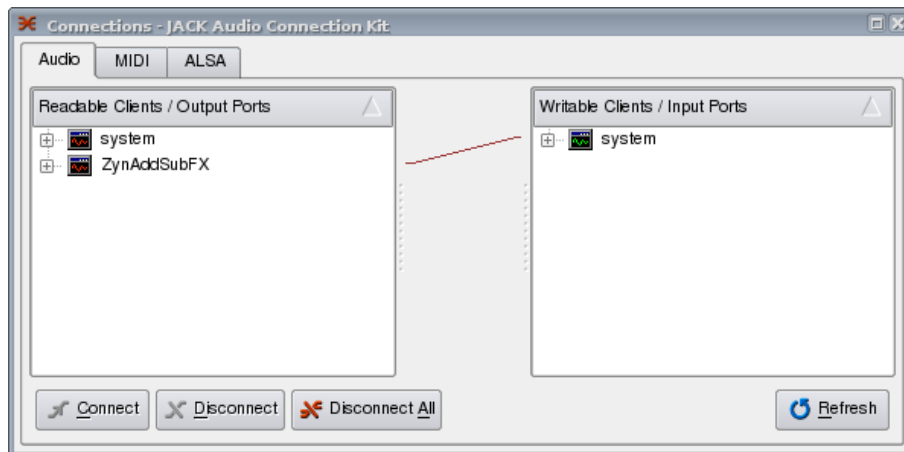
Figuur 3.5: ZynAddSubFX Great Synth!

aan de linkerkant de leesbare (readable) of Output Ports getoond en aan de rechterkant alle mogelijke Audio of MIDI bestemmingen, de beschrijfbare (writable) of Input Ports. Verbindingen kunnen tot stand worden gebracht tussen de bron (Output Ports) en de bestemming (Input Ports) door het selecteren van een Output en Input Port en vervolgens de connect (in het Connections venster) knop aan te klikken. Dit kun je ook doen door het slepen met de muis (rechtermuisknop ingedrukt) van het ene item (Output) naar de bestemming (Input). Het ontbinden van een bepaalde verbinding gaat vrijwel op dezelfde manier, maar nu druk je op de knop Disconnect (hoe logisch). Bestaande verbindingen worden aangegeven door verschillende gekleurde lijntjes tussen de twee panelen (Output/ Input). Elke kleur geeft een verschillende JACK client aan. Het is mogelijk om alle mogelijke verbindingen te leggen tussen de verschillende Audio en MIDI signalen.

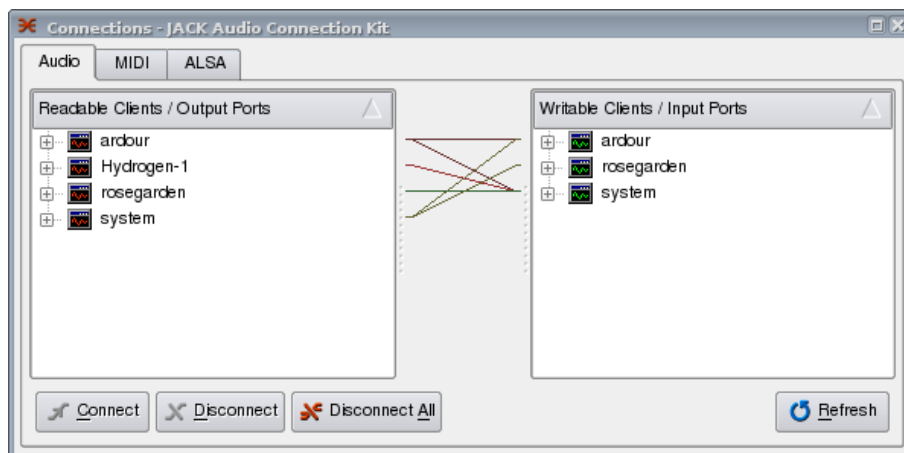
3.2.2 Voorbeeld

Hier een voorbeeld van Hydrogen welke een drumpatroon in Rosegarden afspeelt en wat wordt opgenomen in Ardour (zie figuur 3.7 op de pagina hierna).

En vooruit, nog een voorbeeld van als ik met mijn elektrische gitaar mee wil spelen met een nummer gespeeld door AlsaPlayer. Mijn gitaar gaat via een voorversterker (in mijn geval een simpel Boss distortion pedaal) of mixertje, (capture 1) via mijn 'effectenbox' Rakarrack naar Jackmixer (voor het regelen van de volume) en Jackmixer gaat naar Playback 1 en 2 (monitor/ speaker L en R). De audio van AlsaPlayer gaat ook via Jackmixer naar playback 1 en 2. Via Jackmixer kan ik nu de volume van AlsaPlayer en mijn gitaar regelen (zie figuur 3.8 op pagina 12). (Zie figuur 3.10 op pagina 14 voor een voorbeeld met connecties tussen system,



Figuur 3.6: Jack Connect

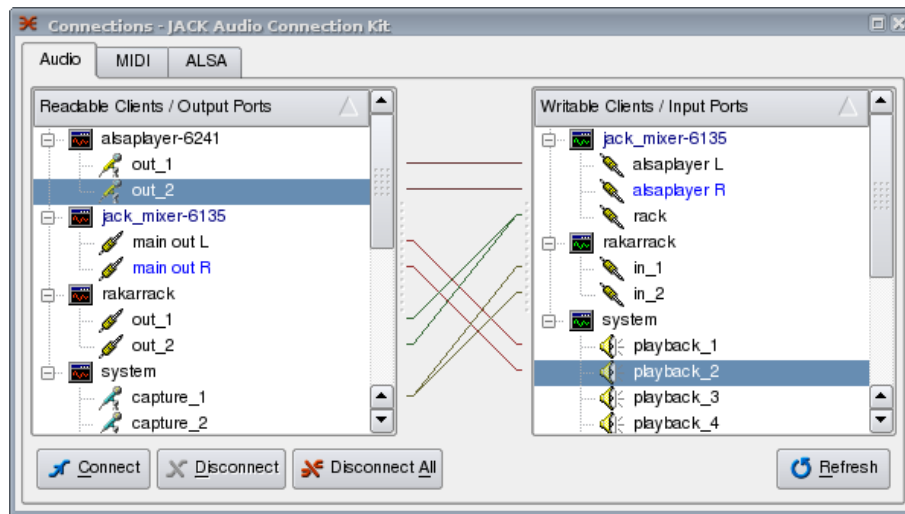


Figuur 3.7: Jack Connect voorbeeld

ardour (3 tracks), hydrogen, whysynth en vkeybd.)

3.2.3 MIDI ALSA

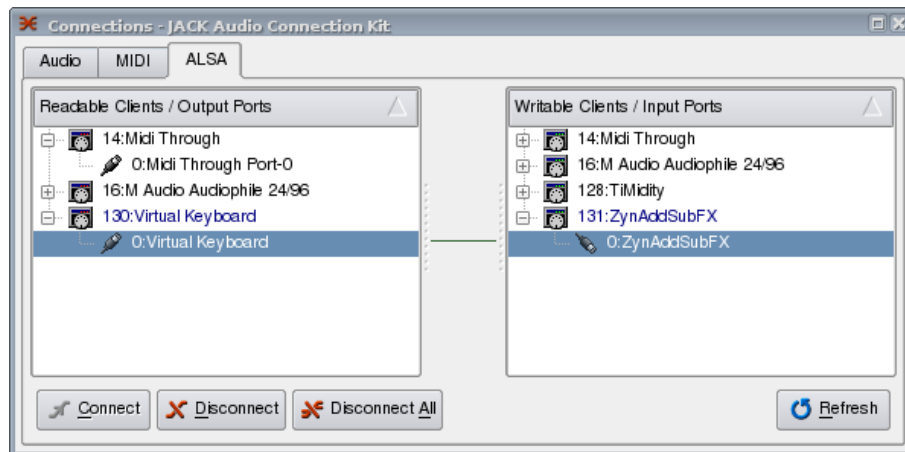
In de MIDI alsa (ALSA) tab in het venster van Connections kun je verbindingen maken tussen MIDI hardware en MIDI software onderling. Hier een voorbeeld van virtueel keyboard (vkeybd) en ZynAddSubFx (zie figuur 3.9 op pagina 13). Als het goed is kun je nu met het virtuele keyboard ZynAddSubFx bespelen.



Figuur 3.8: Jack Connect gitaar voorbeeld

3.2.4 MIDI JACK

Dan is er nog een tabblad over in het Connections venster en dat is MIDI. Dit is voor MIDI software met JACK MIDI support. Als je veel met MIDI werkt, moet je hier maar eens induiken. Applicaties zoals Rosegarden en Qtractor werken naar mijn weten met MIDI ALSA.



Figuur 3.9: Jack Connect ALSA

3.3 De Patchbay

3.3.1 Wat is de Patchbay precies?

Met de Patchbay van Qjackctl kan je connecties tussen Jack clients opslaan en weer openen, waardoor je niets steeds weer dezelfde verbindingen moet maken bij nieuwe sessies of als een sessie onderbroken is. Als de verbindingen verbroken worden door een crash van een programma, dan worden de verbindingen (als de programma's opgestart zijn) automatisch hersteld. De Patchbay kan ook handig zijn bij het verwisselen van instrumenten of een andere setup bij het spelen van verschillende nummers bij een live optreden bijvoorbeeld. Het kost even wat tijd om een Patchbay te maken, maar als je geregeld met Qjackctl werkt dan loont het de moeite!

3.3.2 Een voorbeeld van een patchbay

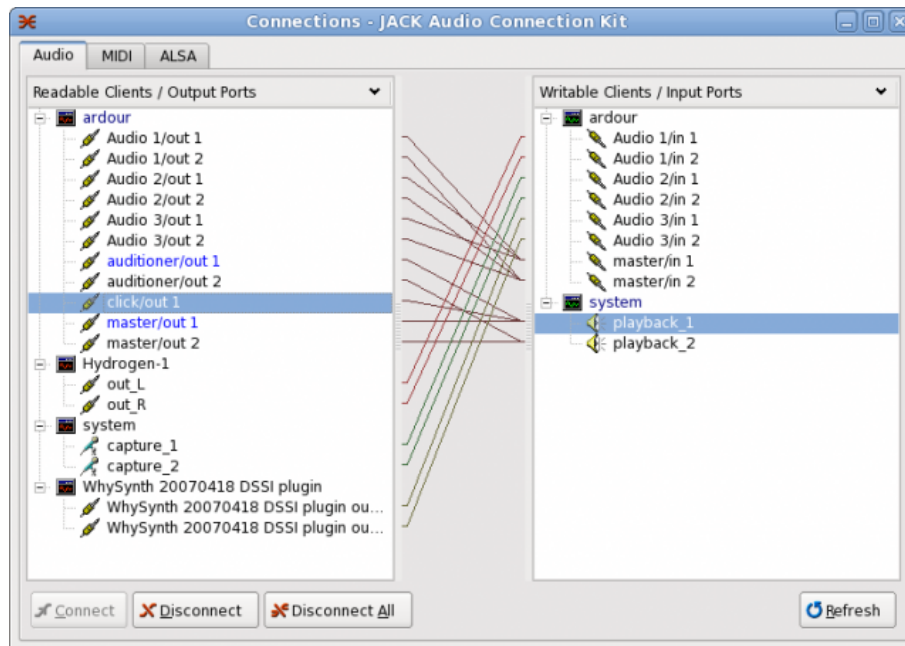
:

In dit voorbeeld maken we gebruik van Ardour, Hydrogen en Whysynth. We zullen een virtueel keyboard (vkeybd) gebruiken als midi keyboard. In figuur 3.10 op de pagina hierna en 3.11 op pagina 15 zie je de connecties die we middels de Patchbay willen maken en opslaan.

3.3.3 Het maken van de Patchbay

1. Start de volgende applicaties:

- qjackctl

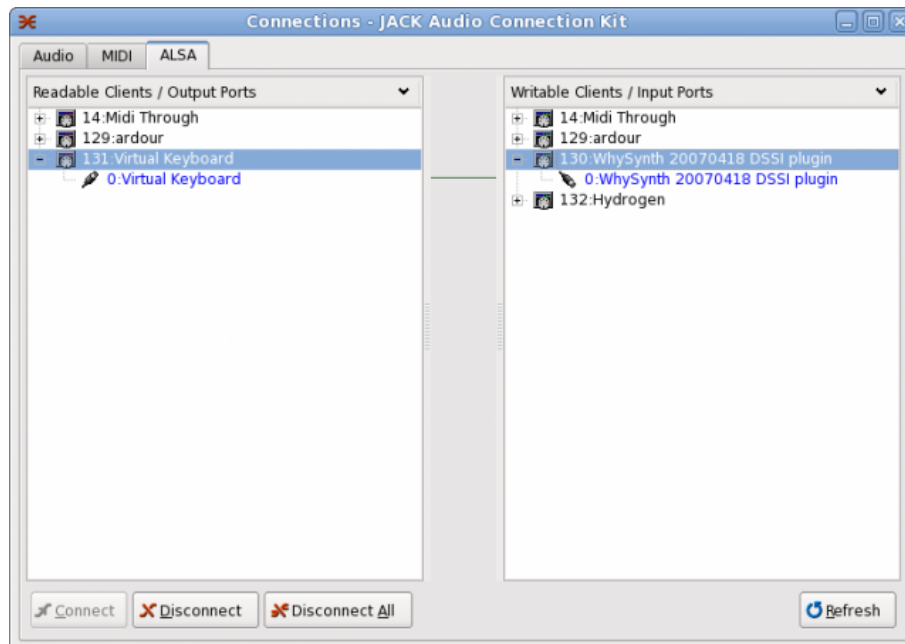


Figuur 3.10: Jack Connections

- ardour (maak drie stereo tracks aan in Ardour)
 - hydrogen
 - whysynth (start door het volgende in de terminal te typen: `jack-dssi-host whysynth.so`).
Meer info: website whysynth
 - vkeybd
2. Verbreek de verbindingen (Disconnect All) van alle audio in het Qjackctl connections venster.
 3. Verbreek de verbindingen (Disconnect All) van alle MIDI (in tabblad ALSA) in het Qjackctl connections venster.
 4. Open de Patchbay (figuur 3.12 op de pagina hierna)

Voor het maken van een nieuw set van Patchbay connections kies ‘new’ (links boven). Als je ‘new’ aan klikt krijg je de volgende vraag (figuur 3.13):

Als je ‘yes’ kiest krijg je connecties met een plug per applicatie en dat is vrijwel zeker niet wat je wilt. Kies dus ‘NO’ om een lege Patchbay te krijgen. De lege Patchbay is opgedeeld in een Input en een Output gedeelte. In de Patchbay worden de JACK clients aangeduid



Figuur 3.11: Jack Connections MIDI (ALSA)



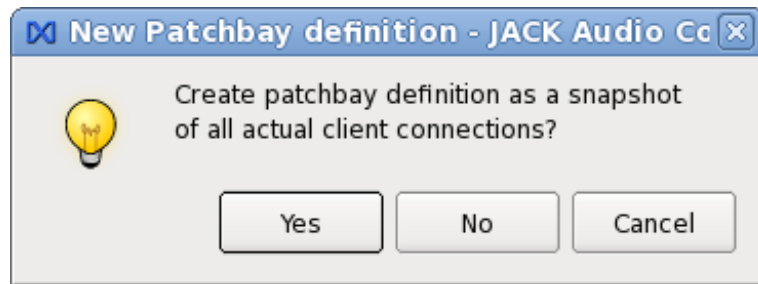
Figuur 3.12: Patchbay button

als sockets. Hoewel de Patchbay Audio en MIDI connecties kan maken, kan er alleen een verbinding gemaakt worden tussen hetzelfde type (Audio of MIDI). We willen voor onze setup een socket voor elke connectie die we gaan maken. Deze sockets zullen voornamelijk stereo zijn (met twee plugs in qjackctl terminologie) of mono (een plug). Indien gewenst kunnen er zoveel plugs per socket gemaakt worden als je wilt...

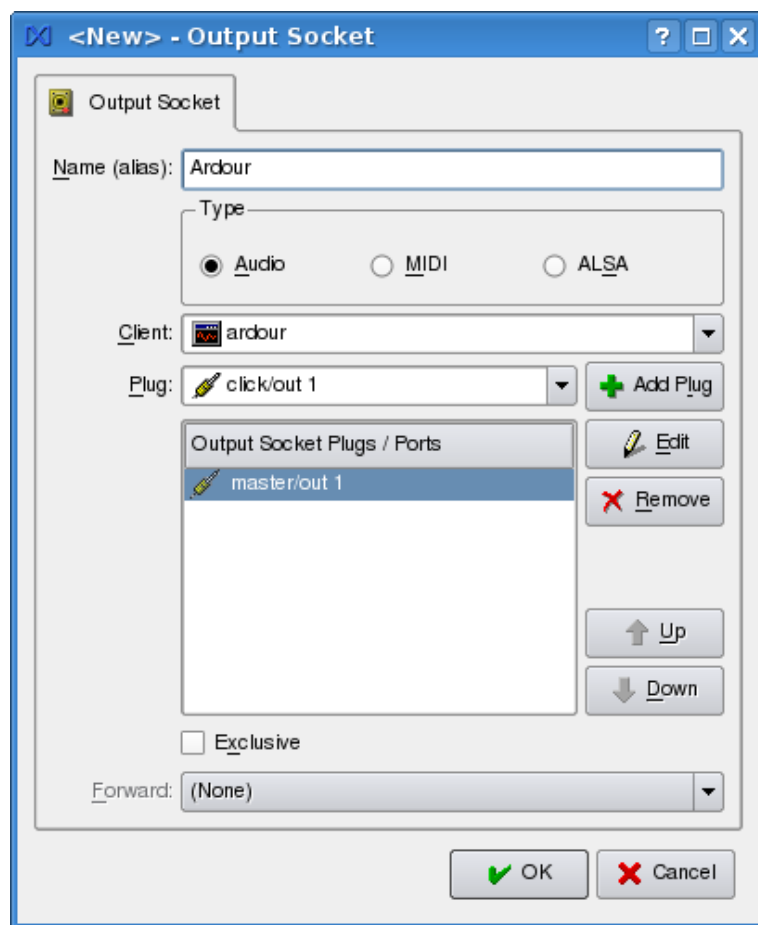
3.3.4 Het toevoegen van Sockets en Plugs

Aan elke kant van het Patchbay venster, zie je een knop 'Add'. Klik op een daarvan en als het goed is zie je dan het venster in figuur 3.14 op de volgende pagina.

We zullen eerst alle output sockets toevoegen en dan alle input sockets, maar je kunt ook per applicatie werken. Wees in ieder geval consistent in je werkwijze. Selecteer nu Audio, Midi



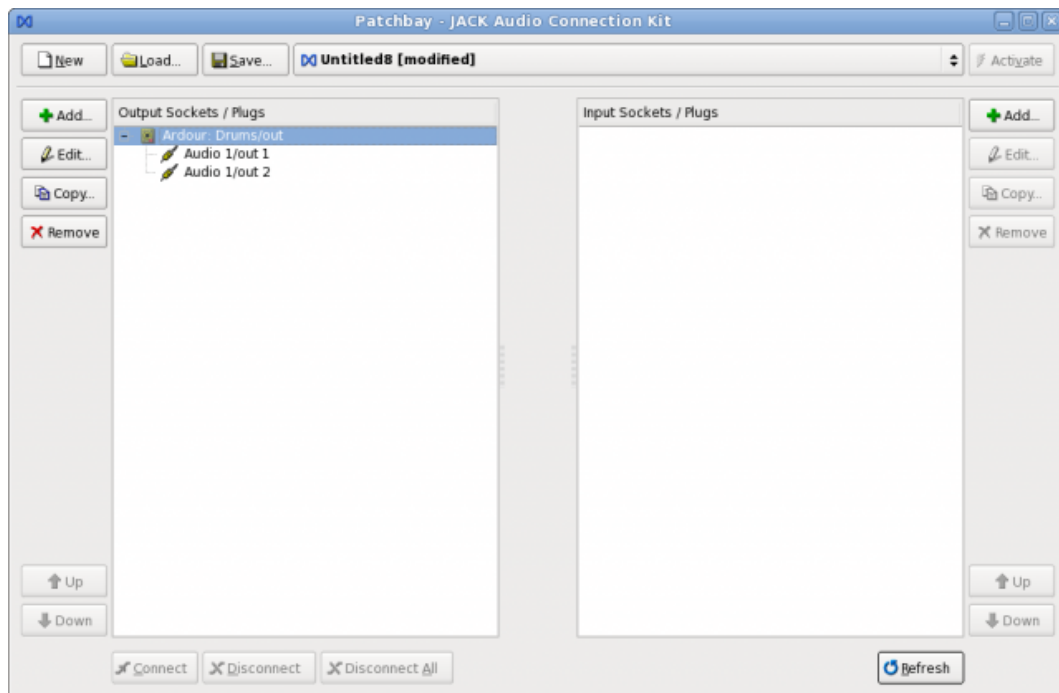
Figuur 3.13: Vraag



Figuur 3.14: Add

Jack of Midi Alsa (ALSA) onder Type. Kies vervolgens een Jack Client. Kies daarna een Plug van de geselecteerde Client. De Plug kun je toevoegen aan je Patchbay door 'Add Plug' aan te klikken. Geef tot slot de socket een naam, zoals 'ardour:audio 1/out' bijvoorbeeld. Als de Patchbay voor een specifieke taak is, zoals deze, dan kun je ook een meer gebruiksvriendelijke

naam kiezen zoals ‘Ardour: Drum track/out’ bijvoorbeeld. Gebruik ‘exclusief’, (wat betekent dat de Patchbay geen andere connecties maakt dan in de actieve Patchbay is weergegeven) en/of ‘forward’ voor de socket als je dat wenst (Zelf ben ik er nog niet helemaal achter waar forward precies voor is.. wie het weet mag het me zeggen). Klik ten slotte op ‘OK’ en je Patchbay moet er dan zo uitzien (figuur 3.15):



Figuur 3.15: Patchbay 1

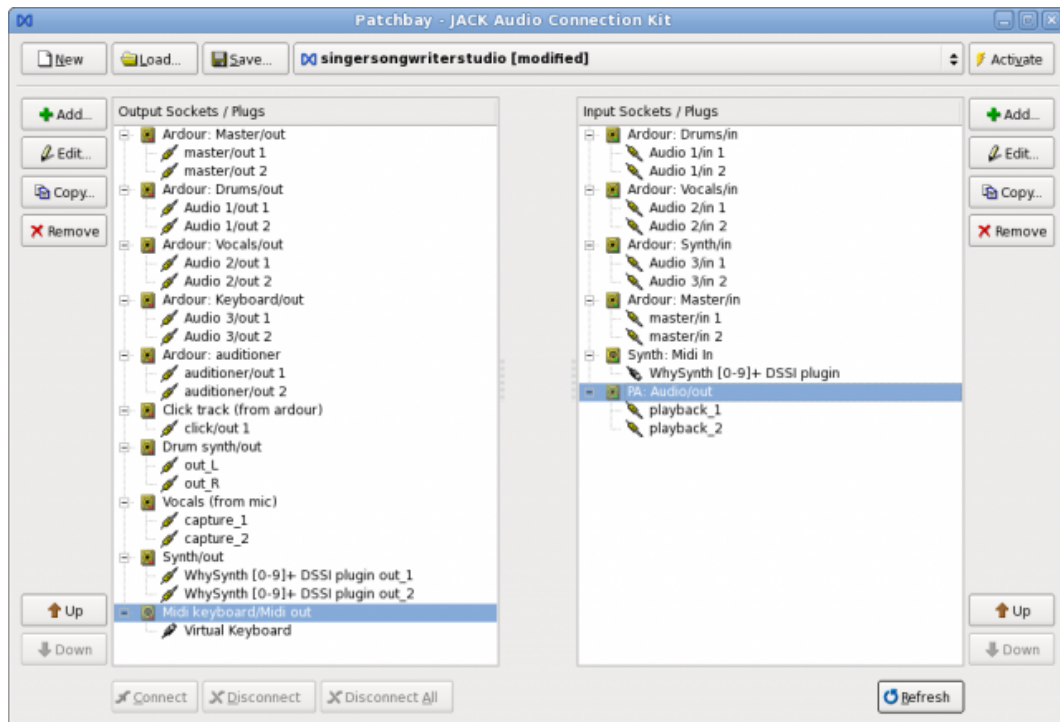
Herhaal dit proces voor elke input en output socket, tot je zoiets hebt als dit (figuur 3.16 op de volgende pagina):

3.3.5 Verbinding maken

Nu kunnen we de connecties maken die we nodig hebben. Selecteer een input socket en een output socket en klik op de Connect knop om ze te verbinden. Herhaal dit tot alle connecties gemaakt zijn en je Patchbay er als volgt uitziet (figuur 3.17 op pagina 19):

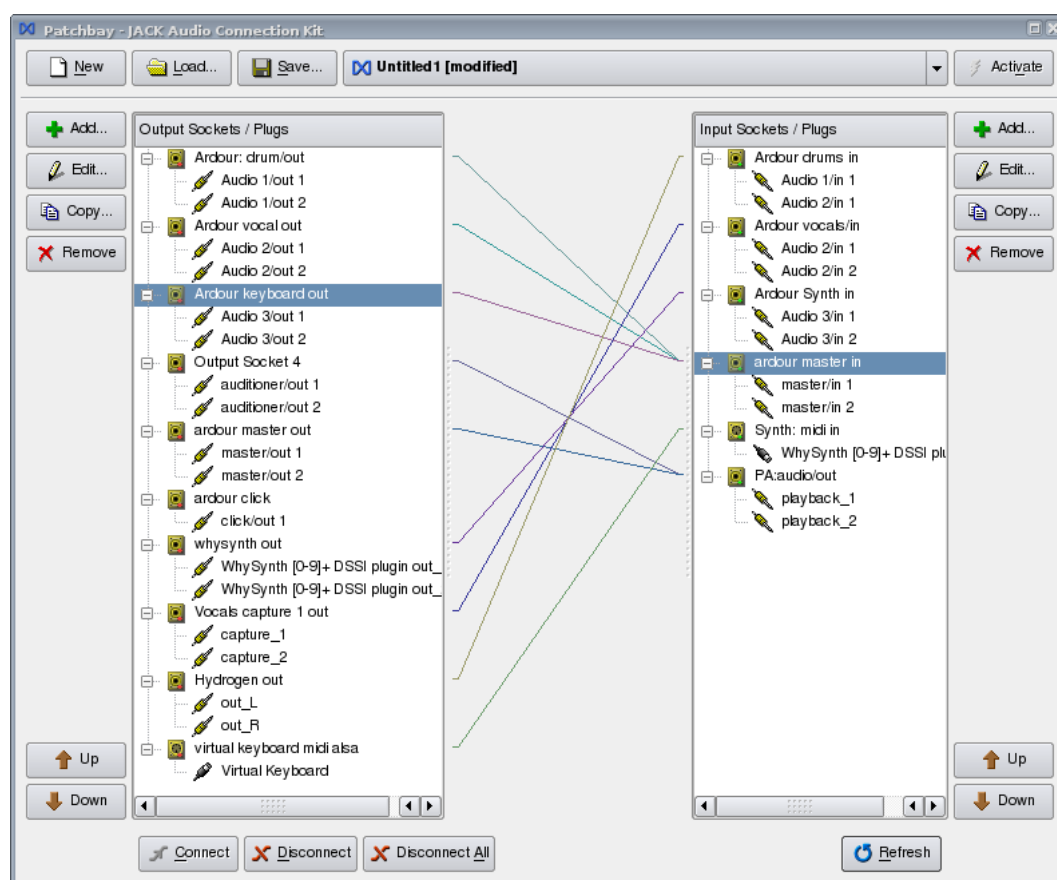
3.3.6 Opslaan en activeren

Je kunt nu de Patchbay opslaan. Klik op de 'activate knop' om de connecties te activeren. Als het goed is zie je nu de gewenste connecties onder Connect van Qjackctl. Om de Patchbay te testen kun je onder Connect alle connecties verbreken (Disconnect all), klik op ignore om de connecties niet uit de Patchbay te halen (ok als je dat wel wilt) en kijk of na het verbreken



Figuur 3.16: Patchbay 2

de connecties weer automatisch gemaakt worden. Als je de volgende keer met dezelfde sessie wilt werken, dan activeer je de gemaakte patchbay en open je de programma's, die zich automatisch volgens de regels van de patchbay zullen verbinden. Handig niet?



Figuur 3.17: Patchbay eindversie

3.4 Jack synchronisatie

Zoals reeds in het begin van dit artikel is vermeld is een belangrijke functie van JACK ook het synchroniseren (gelijktijdig afspelen) van de verschillende JACK clients. In de wereld van audio en muziek software wordt met synchronisatie 'de precieze timing van audio processen (events) met video, MIDI, en andere soorten 'streams', genoemd.

3.4.1 JACK Transport Control System

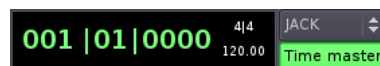
De mogelijkheden van het managen van een complex aantal verschillende audio in en outputs (I/O) door JACK, is niet veel waard gedurende een opname of het beluisteren als de verschillende tracks, als de channels en streams niet qua tempo en timing precies op elkaar afgesteld zijn. Elke muzikant weet hoe belangrijk het is om in de maat de spelen! Om te zorgen dat de verschillende Jack clients samen in de(zelfde) maat spelen en allemaal op het juiste moment beginnen is er een transport control subsystem' aanwezig in JACK welke met QjackCTL bedient kan worden. Het systeem zorgt voor een 'sample-accurate' synchronisatie

van de streaming audio I/O en elke Jack client met Jack transport support (!), kan bediend worden door elke andere Jack client die ook Jack transport support heeft. Dit noemt men ook wel de 'master Client', maar in principe kan elke applicatie met Jack transport functionaliteit als 'master' functioneren. Het tempo van de transport (het muzikale tempo (bpm)) wordt door een van de Clients (de Master) bediend, zodat de andere applicatie op het zelfde tempo afgespeeld wordt. Je hoeft niets extra's te doen om het 'Jack transport control system' te installeren of op te starten. Dit gebeurt automatisch als je Qjackctl opstart.

3.4.2 Voorbeeld: Hoe Hydrogen synchroniseren met Ardour

In dit voorbeeld laten we zien hoe je Ardour en Hydrogen (drum machine) kunt synchroniseren met behulp van JACK en Qjackctl. Dit maakt het mogelijk om een drumpatroon af te spelen in perfecte timing met een sessie in Ardour.

- Start Qjackctl
- Start Ardour en creat a new session.
- Zorg dat de Time Master aan staat (groen) en op JACK staat in plaats van Internal.



Figuur 3.18: Time master

- Als je dat gedaan hebt, open dan Hydrogen.
- Open een demo ↵ File ↵ open demo
- Voordat je naar Ardour gaat, zorg dat JACK TRANSPORT aan staat in Hydrogen (licht blauw) .



Figuur 3.19: Hydrogen sync

- Maximaliseer Ardour en druk op de Play knop. Als het goed is hoor je nu hydrogen spelen.
- Je kunt ook QjackCTL gebruiken om de applicaties te starten. Klik dan in QjackCTL op de play knop.



Figuur 3.20: QjackCTL JACK Transport Start

3.4.3 Vele mogelijkheden

Bovenstaande is slechts een voorbeeld. Synchronisatie is mogelijk met veel andere applicaties als Rosegarden, Muse en Seq24.

Hoofdstuk 4

Achtergrond informatie

- Website JACK
- Website QjackCTL
- Linux Musicians Wiki

4.0.4 Met dank aan

- JACK is een product van Paul Davis en anderen. Paul is ook de auteur van Ardour. Ga voor donaties naar Ardour.org
- QjackCTL is een product van Rui Nuno Capella. Ga voor donaties naar website Rui Nuno Capela.

4.0.5 Referenties

Dit document is voor een aanzienlijk deel een vertaling van werk van anderen. Ook zijn er enkele afbeeldingen gebruikt uit stukken van anderen. Dit zijn de bronnen die gebruikt zijn.

- Linux Musicians Wiki
- 64studio.com
- Jacklab.org
- Ubuntu: ubuntu studio blog
- HowTO by Simon W. Fielding and Rui Nuno Capela about Patchbay
- Article by D.Philips about Jack Sync