www.rncbc.org/drupal/node/76

rncbc.org
a.k.a. Rui Nuno Capela

# QjackCtl and the Patchbay

10 March, 2008 - 14:06 — rncbc

Simon W. Fielding writes that after quite some time, and reading between the lines of various posts on the linuxaudio-user mail list, he has finally understood how the QjackCtl Patchbay works and what it is for. Because of that, he has written a brief guide to using the patchbay and the document is here first released into the wild, hoping it will turn out to be useful to someone in the quest to make amusing noises at home.

Thanks Simon, for this excellent piece of work. You've done, in just a few paragraphs, what I've failed to do in all those four years QjackCtl's Patchbay has been around ;) Cheers.

## What is this document for?

When I started using qjackctl I couldn't understand the difference between the patchbay and the connections window. They both seemed to show the same information and the patchbay was not very useful. Now, thanks to various hints on the linuxaudio mailing list, I think I understand how it should work and how useful it could really be. This document is my attempt to share that understanding with those in the same position as I was.

It is not an explanation of how to setup jack, qjackctl or any other tools and assumes that you can already do that but I hope it will be useful to somebody nevertheless.

## What is the patchbay for?

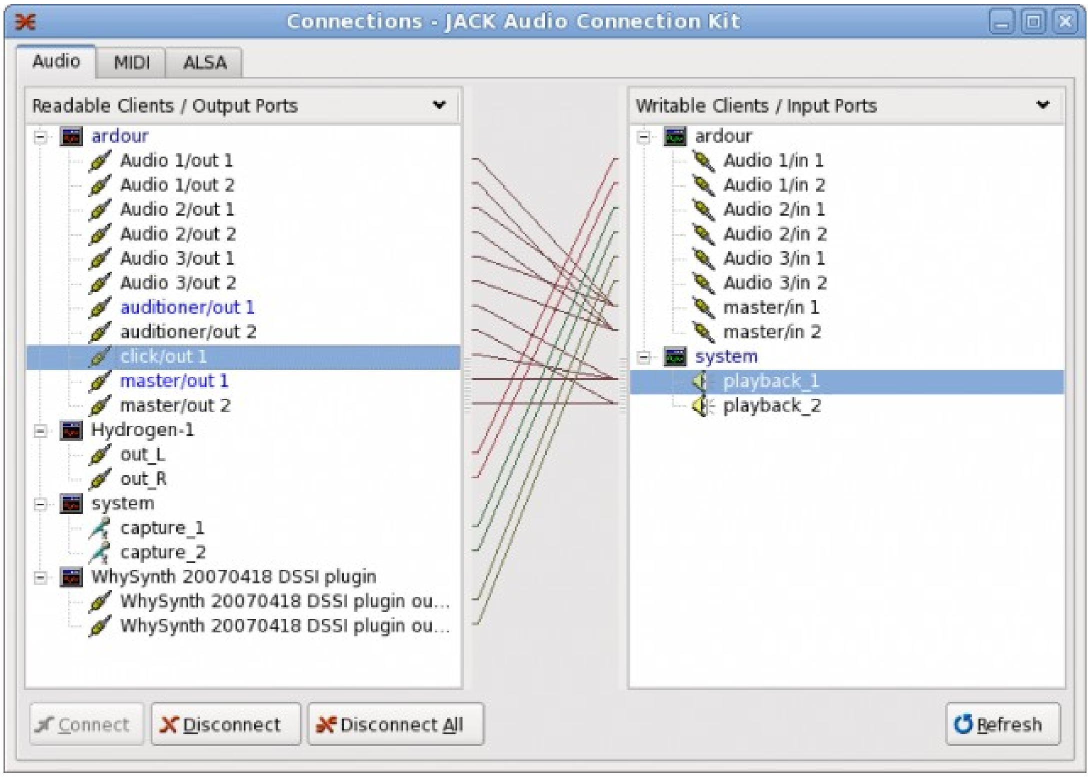
When it is setup correctly, the patchbay provides automatic connection setup for jack-aware applications, maintains connections persistently through application stops (or crashes) and restarts, and allows for easy switching between different setups (eg from one instrument setup to another between songs in a live performance).

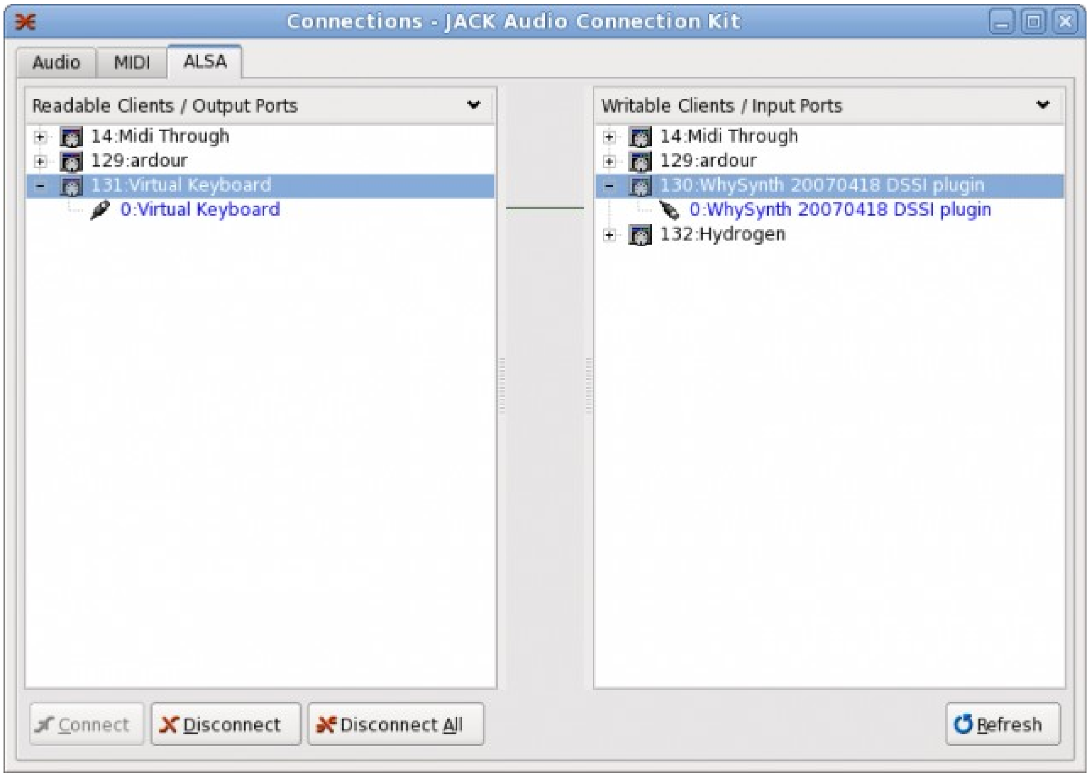
## What shall we do today?

Since I am far too shy to stand up and perform live in front of an audience, I will attempt to describe how to use the patchbay to implement a simple studio rig which I can use to record drums, keyboard track and a single vocal.

www.rncbc.org/drupal/node/76

I will use ardour, hydrogen and whysynth as the tools to be connected. Since I am writing this away from my studio, I will use vkeybd to stand in for my real midi controller keyboard. The final jack connection setup which we are aiming for is shown in the images below.

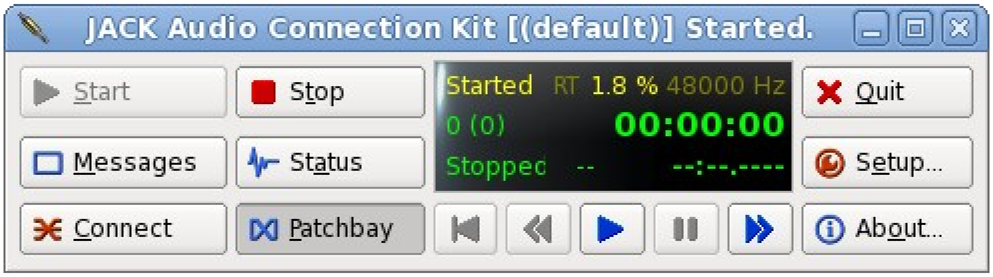



Connections - JACK Audio Connection Kit
Audio
MIDI
ALSA
Readable Clients / Output Ports
14:Midi Through
129:ardour
131:Virtual Keyboard
0:Virtual Keyboard
Writable Clients / Input Ports
14:Midi Through
129:ardour
130:WhySynth 20070418 DSSI plugin
0:WhySynth 20070418 DSSI plugin
132:Hydrogen
Connect
Disconnect
Disconnect All
Refresh

www.rncbc.org/drupal/node/76

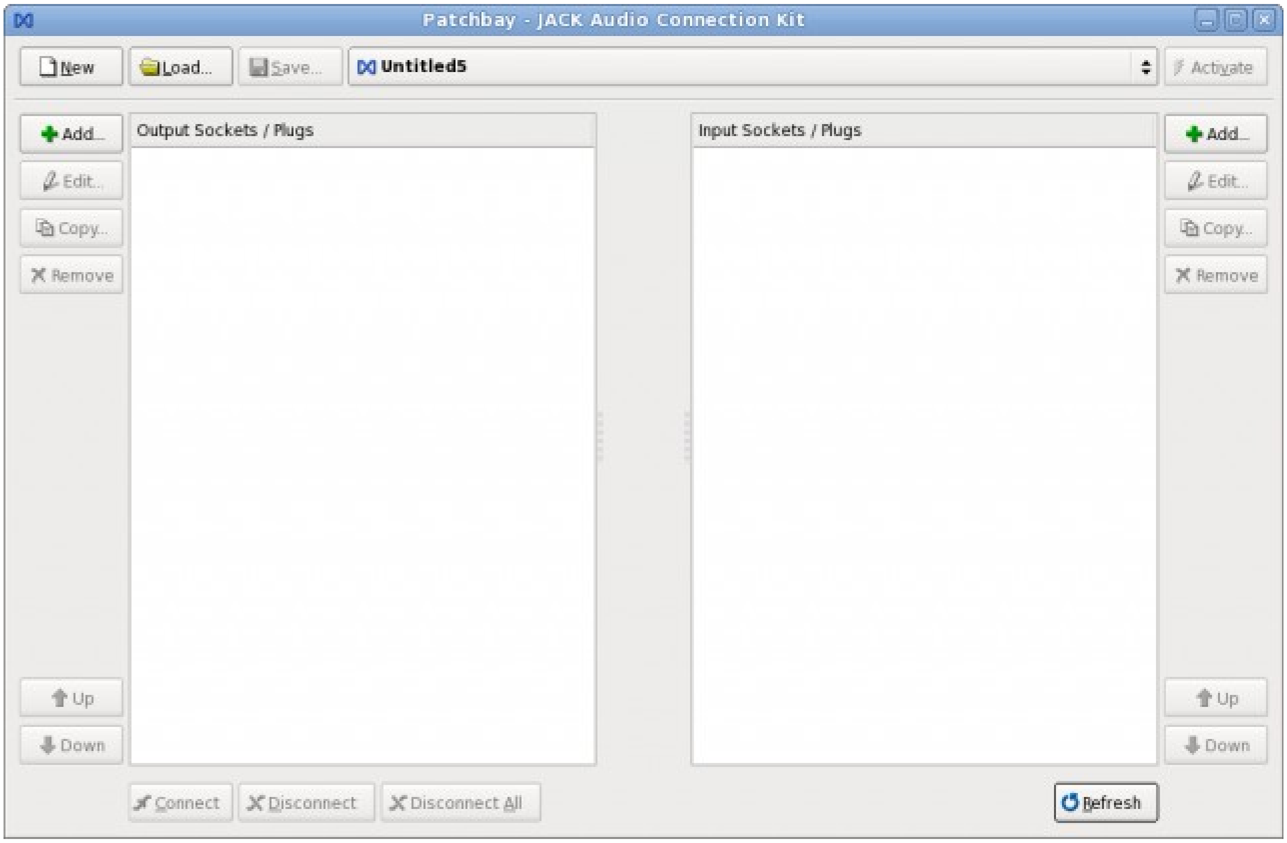
# How do we do it?

1. Start all your applications
   - qjackctl (and then start jackd)
   - ardour
   - hydrogen
   - whysynth
   - vkeybd
2. Disconnect all audio in qjackctl Connections windows
3. Disconnect all midi in qjackctl Connections windows
4. Open the Patchbay



JACK Audio Connection Kit [(default)] Started.

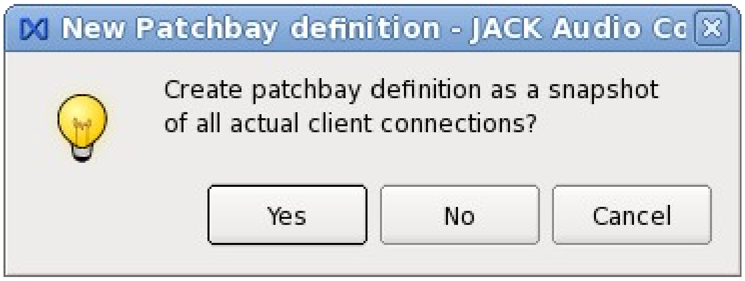
```
Started  RT 1.8 % 48000 Hz
0 (0)           00:00:00
Stopped  --     --:--.----
```



Patchbay - JACK Audio Connection Kit

Untitled5

At this point, if this is your first set of patchbay connections, the system is ready for us to start adding plug definitions. However, when you have finished and wish to create another patchbay definition for the next song/overdub/band in your studio, you will need to click on the "New" button. This is where all my confusion and problems started because I assumed that you needed to start by clicking "New" so I will explain what not to do. When you see the dialog box which asks if you wish to take a snapshot :-



New Patchbay definition - JACK Audio Co

Create patchbay definition as a snapshot
of all actual client connections?

do **NOT** click "Yes". This will give you one plug per application which will almost certainly not do what you want if you connect them. Either click "No" to get an empty patchbay, or "Cancel" to give up and do something else instead.

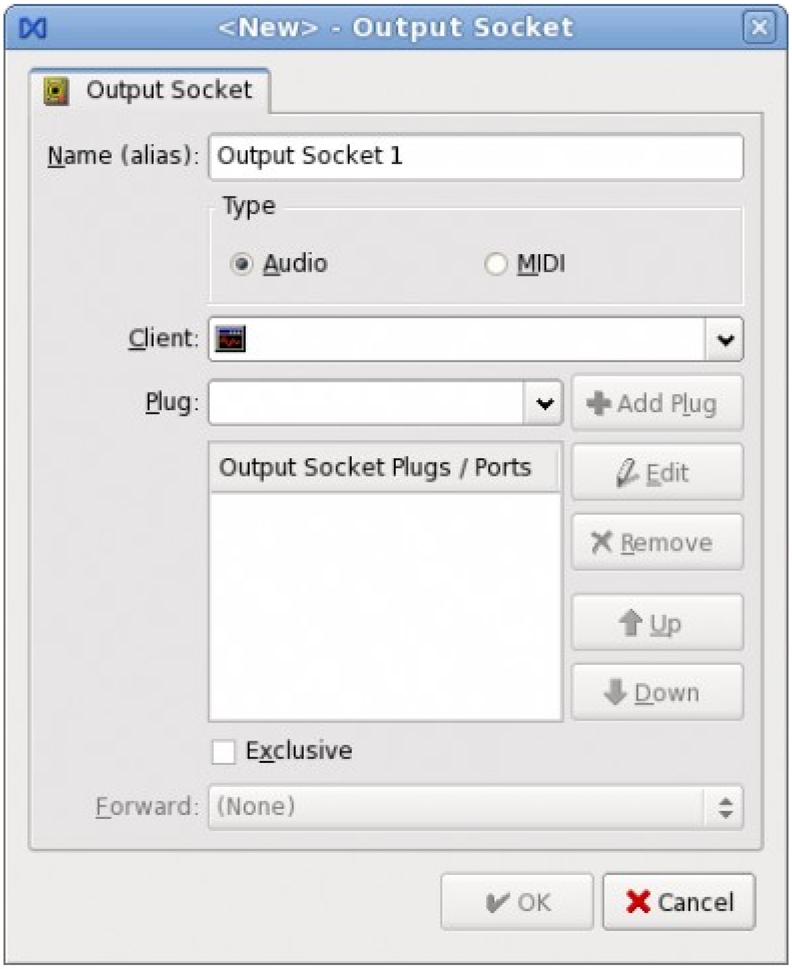
Assuming you didn't give up, you now have an empty patchbay, split into Input and Output sections. This is where we start the real, interesting part. It can be a bit tedious but we should only have to do this once so it is worth persevering and getting it right.

What I want for my setup is to have one socket for each end of each connection I want to be able to make. These sockets will be mostly stereo (ie with two plugs in qjackctl terminology) or mono ( ie one plug). There is no need to restrict yourself however and you could connect as many plugs to each socket as you wish (caveat - I don't actually know if there is a limit or what it is but it's more than is likely to be needed in any manageable audio setup).

## What now?

On each side of the patchbay window, you will see a button labelled "Add". Click one of them and you will see the window below. It is best to be methodical about this stage. I tend to add all output sockets and then all input sockets but you may prefer to setup all sockets for a single application and then move on to the next. Just do what works for you but try to be consistent.

www.rncbc.org/drupal/node/76



Name (alias): Output Socket 1

Type: ● Audio ○ MIDI

Client:

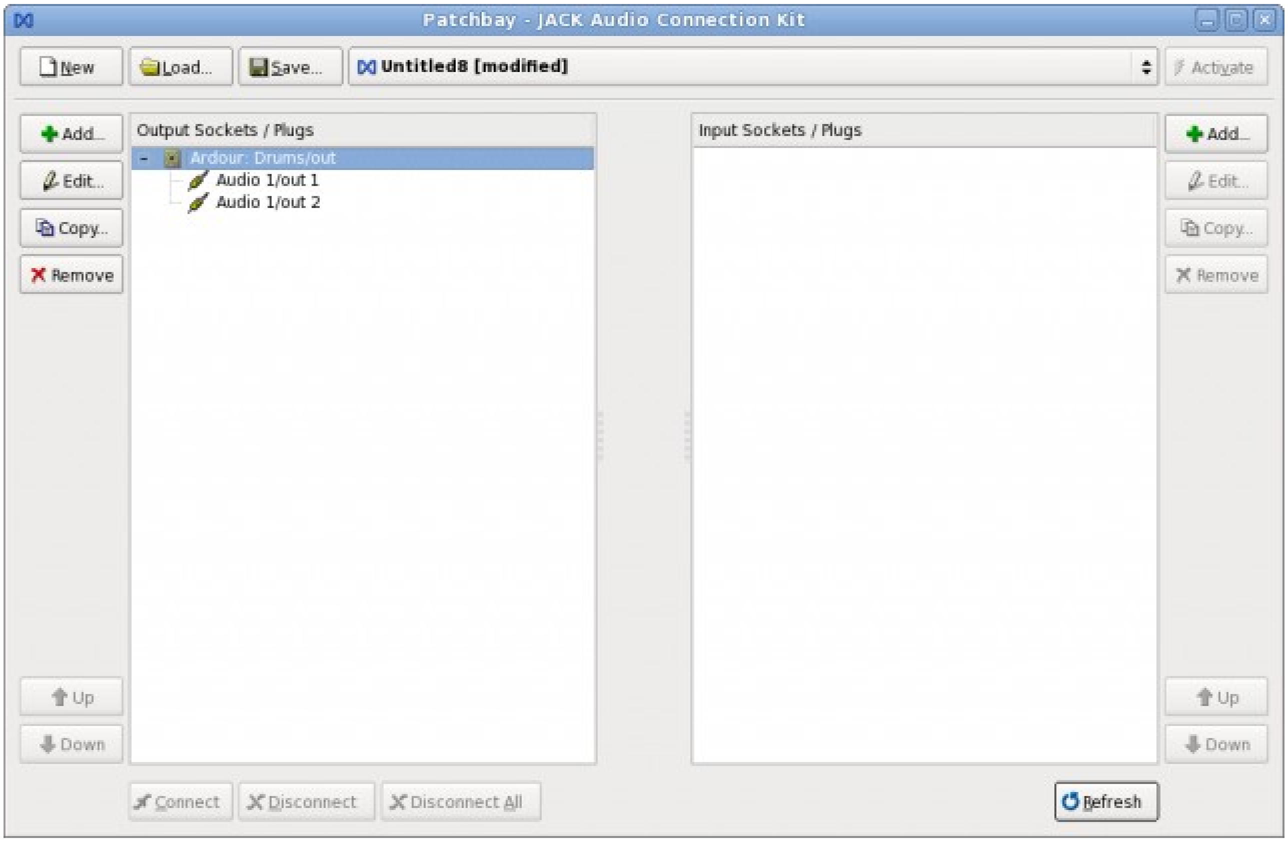
Plug:

□ Exclusive

Forward: (None)

Now select Audio or Midi using the radio buttons, select your client from the drop down list, then select the plugs you want to add and click "Add Plug". Finally, give the socket a name that will mean something to you when you come to use the patchbay later on. If it's a general purpose patchbay, you might want to use something like "ardour: Audio 1/out" for example. If the patch bay is for a specific task however, as this one is, then you might want to give it a more user-friendly name eg "Ardour: Drum track/out". The name can be anything you want it to be. Use "exclusive" and/or "forward" on the socket if you wish. Finally, click "OK" and your patchbay should look like this :-

www.rncbc.org/drupal/node/76



Untitled8 [modified]

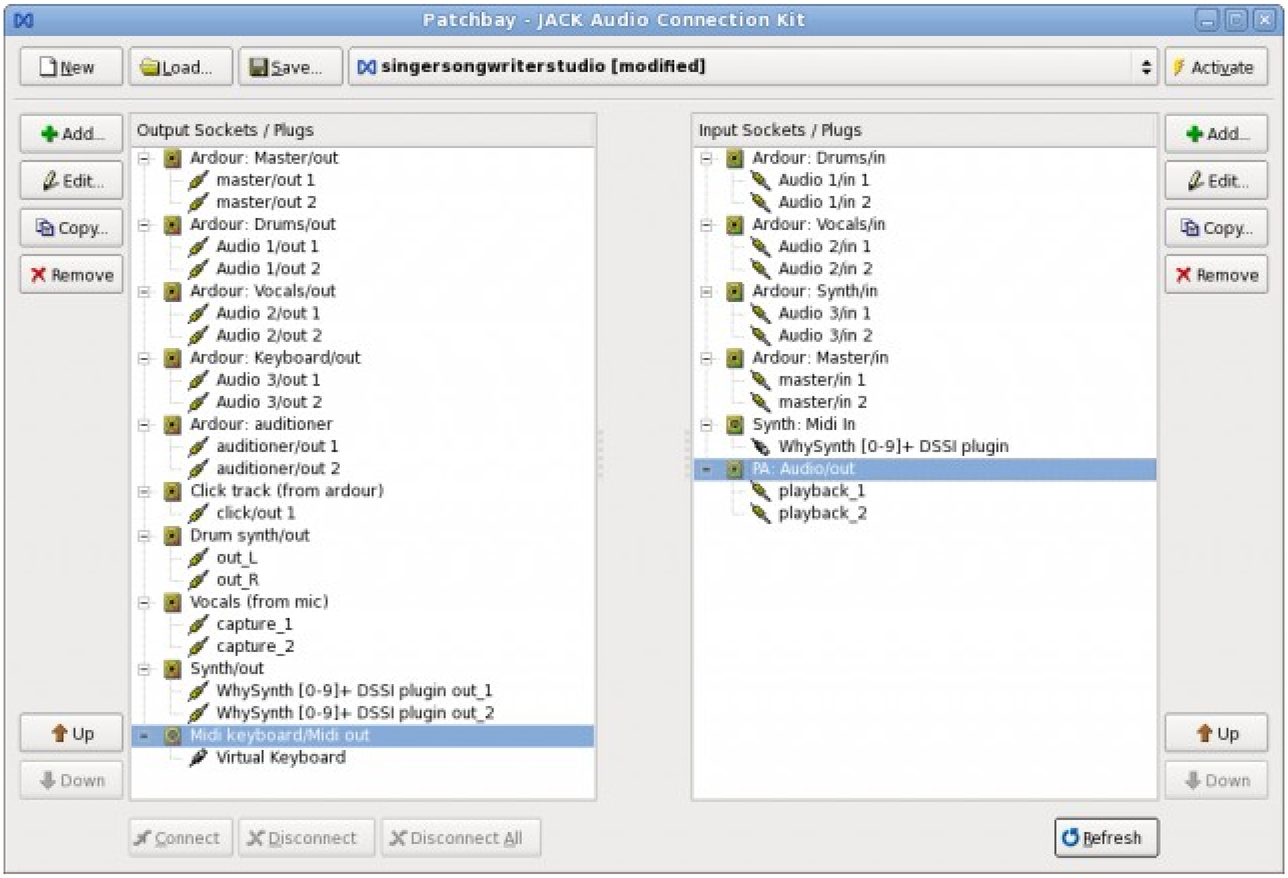
Output Sockets / Plugs

- Ardour: Drums/out
  - Audio 1/out 1
  - Audio 1/out 2

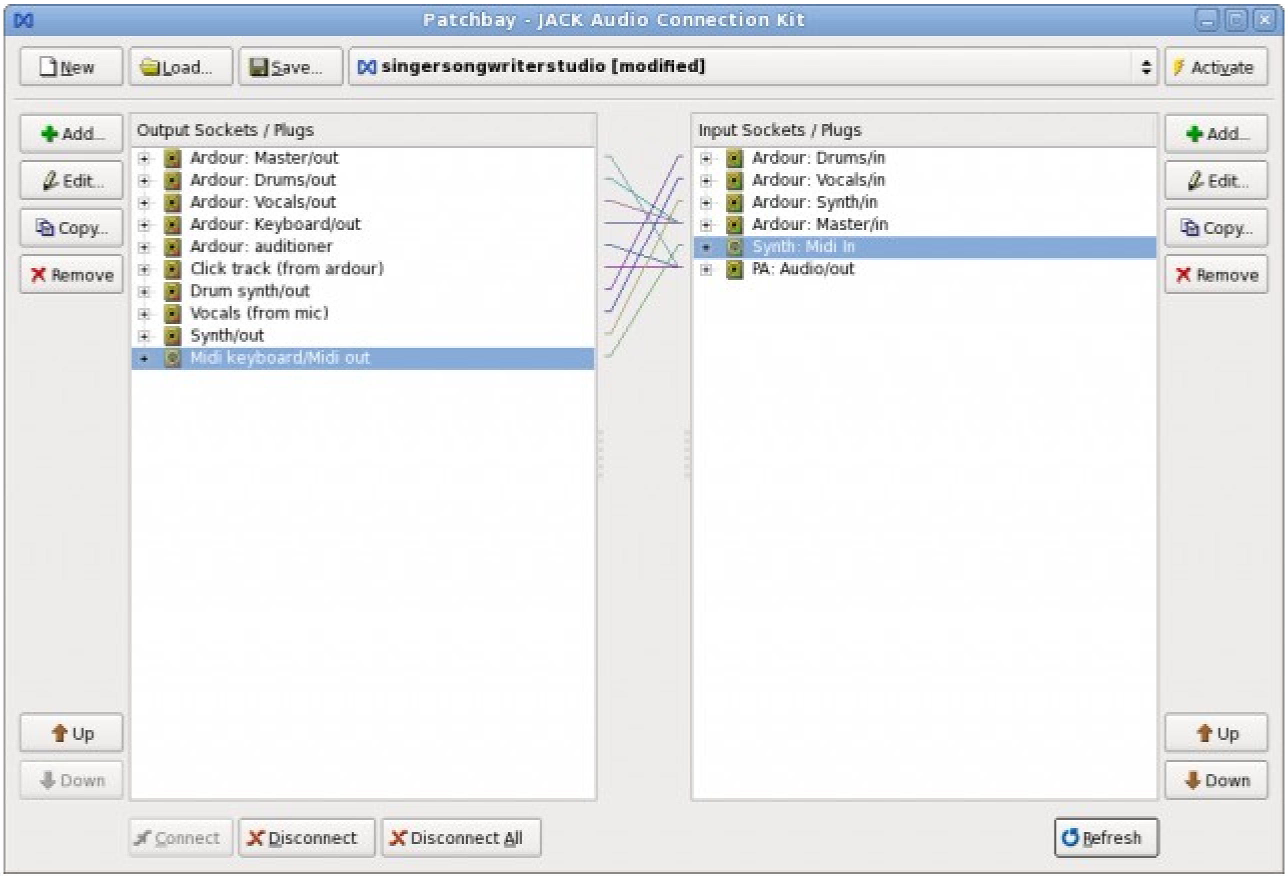
Input Sockets / Plugs

Now repeat this process for each input and output socket you wish to create until you have something like this :-

[www.rncbc.org/drupal/node/76](http://www.rncbc.org/drupal/node/76)



Now we can make the connections that we need. Simply select an input socket and an output socket and click the Connect button to connect them together. Repeat until all connections are made and your patchbay looks something like this :-



singersongwriterstudio [modified]

Output Sockets / Plugs

- Ardour: Master/out
- Ardour: Drums/out
- Ardour: Vocals/out
- Ardour: Keyboard/out
- Ardour: auditioner
- Click track (from ardour)
- Drum synth/out
- Vocals (from mic)
- Synth/out
- Midi keyboard/Midi out

Input Sockets / Plugs

- Ardour: Drums/in
- Ardour: Vocals/in
- Ardour: Synth/in
- Ardour: Master/in
- Synth: Midi In
- PA: Audio/out

Now, save the patchbay definition, give a memorable name, and click the activate button.

When you have done all this, go back to the qjackctl connections window and we should see all the applications connected as we have specified. To test the patchbay, click "Disconnect All" in the connections window and watch all the connections disappear and then automagically reappear.

## What does the patchbay NOT do?

The patchbay does **not** disconnect audio or midi connections. If you disconnect one, or all, sockets in the patchbay, all the audio and midi connections will still be there. The audio/midi connections will still automagically reappear if you disconnect them until you *save* the (now disconnected) patchbay and subsequently disconnect the audio/midi connection in the connection window.

*Simon W. Fielding*
s.fielding at wmcce dot org

- rncbc's blog
- Add new comment

# Comments

14 March, 2008 - 11:47 — rncbc

## QjackCtl Connections vs. Patchbay

The following verbosity intends to be one's formal sidenote regarding the distinction between QjackCtl Connections and Patchbay widgets, while quite similar in their aspect, aren't at all fitting the same purpose nor functionality. Let's just hope this makes it a bit more bearable ;)

**QjackCtl Connections**

All audio and MIDI connections and routing are controlled under the QjackCtl Connections window. These are flexible but imperative connections that can be made between all the Audio/MIDI software application programs (clients) that run in the system, establishing an actual signal path from source to destination ports.

In the Connections window you will see all Audio/MIDI signal sources on the left pane (Readable Clients/Output Ports) and all possible Audio/MIDI signal destinations in the right pane (Writable Clients/Input Ports).

Actual connections are established between source (Output Ports) and destination (Input Ports) by selecting one item in either side and pressing the [Connect] button at the bottom-left of the Connections window.

It is also possible to establish a connection by dragging an item and dropping it over the other side (Note: it doesn't matter from either side you start dragging).

To disconnect a currently established connection you proceed in a similar manner: select each item in either side and press the [Disconnect] button.

Actual and established connections are shown as color directed lines in the middle pane of the Connections window. Each color indicates a single and distinct source module (client) that is currently connected.

It is possible to route every audio and MIDI from and to where you see fit, provided that JACK and ALSA sequencer services and client applications are up and running.

**QjackCtl Patchbay**

All connections made in the Connections interface are kept as long you don't power-cycle the JACK server (jackd). That is, all connections will be lost when the JACK server or any of the client application programs are closed or terminated. Those connections just don't get setup whenever the server and any of its client applications get started later again.

www.rncbc.org/drupal/node/76

To maintain a custom and persistent connection configuration layout, you'll have to edit and activate a ***patchbay definition*** profile. Patchbay definition profiles are connection models that are edited and created in the Patchbay window, which is accessed via the [Patchbay] button on the main QjackCtl panel.

The Patchbay window is very similar to the Connections one. However, it is here where you prepare and set your intentional and persistent connections, not the actual connections. When activated, this patchbay definition profile will keep all declared connections automatically, as long as QjackCtl is kept alive.

Following the Patchbay model, you declare connections between ***sockets*** (clients). When two sockets are said connected, it is implied that each one of its ***plugs*** (ports) are declared about to be connected in turn, one by one, in a one-to-one basis.

Take special note on the plug/port order in the socket plug list, as each plug of each other socket is about to be connected in the precise order those are listed on either end.

The Patchbay handles both Audio and MIDI connections. However you can only instruct to connect between sockets of the same type.

Each actual Audio/MIDI client program is mapped into one or more items in a patchbay definition, here called sockets. Thus far, a socket is just one representation to a specific subset of all the ports (plugs) of a given client.

For example, you can have a stereo socket on to which only two of the ports from a given client are listed. When you refer to this socket you are actually referring only to those ports. Take special care that you'll have to agree to yourself to always list the ports in the very same and logical order, e.g. Left always followed by Right channel, which is the most common.

You can set a socket as being in ***Exclusive*** mode, meaning that the patchbay routine will enforce that no connections are to be held at any one time, other than the ones explicitly declared by the active patchbay definition.

As an additional feature, you can also set one socket to ***Forward*** and therefore replicate all its connections to yet another one. This option feature may be found useful, for instance, when you wish to replicate all connections made onto the main audio outputs to some aux or cue (eg. phones) outputs.

Hope it helps a little bit.
--
rncbc aka Rui Nuno Capela