

Linux-based audio recordings

Hello? Is this thing on?

Matthew Geddes
matt@musicalcarrion.com

June 2006

Overview

- ▶ Quick overview of my studio
- ▶ Including a particular project
- ▶ Not the only way to do the same thing

Scope

- ▶ Focus more on the process and the role Linux played (and plays)
- ▶ Less focus on audio engineering stuff – plenty elsewhere
- ▶ Specific example session used to illustrate
- ▶ Specific tools used comes down to preference in many cases

The project - The Upside Down

- ▶ A duet currently based in Adelaide
- ▶ Talented musicians and songwriters
- ▶ Guitarist (Elle Taylor) and Vocalist (Gini Herrick)
- ▶ Very heavily influenced by old blues and modern alternative rock
- ▶ Were a little disappointed with previous attempts to capture their music
- ▶ Often found on Tuesday nights playing at Higher Ground
- ▶ Know them before they become famous ;-)

Planning

Always plan first and make plenty of notes:

1. Run through a bit of each track to be attempted
2. Is a given track better suited to multitrack or live?
3. Instrumentation (and effects)
4. Order in which to record
5. Get your levels right!

Track one - Wishing Stone

- ▶ Very bluesy track
- ▶ Well-suited to live jam recording
- ▶ Wood-bodied resonator guitar for rhythm
- ▶ Steel-bodied resonator guitar for lead/slide
- ▶ Recorded in small room with particular reverb characteristics
- ▶ Two microphones pointing at vocals and rhythm guitar
- ▶ Three takes (all kept)

Track one - post processing

- ▶ Minimal processing done, due to live recording style
- ▶ Used a little software compression to remove peaks
- ▶ Software amplification / normalisation
- ▶ Nice recording deliberately destroyed with LADSPA effect *Lo-Fi*

Track two - Fuselage

- ▶ A slightly more *rock* feel
- ▶ Suited to multitrack
- ▶ Fender Squier Telecaster through DI box
- ▶ Close vocal mic
- ▶ No drums due to time constraints (Hydrogen considered)

Track two - post processing

- ▶ Some (too much?) reverb added to vocals
- ▶ Overdubbed part of guitar track
- ▶ Compression and normalisation (vocal and master)
- ▶ Manual compression using -amplify and EQ
- ▶ Panning corrections

Track three - Behind These Walls

- ▶ Similar style to previous track (hence order)
- ▶ Same guitar / FX combination
- ▶ Close vocal mic
- ▶ No drums

Track three - post processing

- ▶ Better amount of vocal reverb for space and sparkle
- ▶ Compressed vocal track a little
- ▶ Manual compression using -amplify and EQ
- ▶ Panning corrections
- ▶ Mild compression and normalisation

Hardware used

- ▶ \$1000AUD Acer laptop + M-Audio Audiophile USB
- ▶ Apple PowerBook
- ▶ Assortment of microphones
- ▶ DI box (Line6 POD)
- ▶ Cheap 12 channel mixer
- ▶ Noise-making devices (guitars, bitter ex-girlfriends, etc)

Hardware signal flow



Brand names

- ▶ There is no *best* Linux audio application
- ▶ There are a lot of great tools out there (such as....)
- ▶ There is plenty of overlap in functionality
- ▶ Use the right tool for your situation
- ▶ Poke around `linux-sound.org`
- ▶ ALSA vs OSS

Acer paperweight

- ▶ Primarily used to capture initial audio tracks
- ▶ Running Debian 3.1 with custom 2.6 kernel patched for low-latency
- ▶ Latest ALSA drivers package
- ▶ JACK 0.100.1 with magical byte-swap package
- ▶ ecasound 2.4.3
- ▶ LADSPA

Apple G4 PowerBook

- ▶ Mac OS X 10.3.x
- ▶ Used primarily for mastering final stereo tracks
- ▶ Used due to mixing being done in a cafe (battery life)
- ▶ Running JACK OS X version 0.100.1
- ▶ ecasound 2.4.4
- ▶ Audacity 1.2.2 (for minor manual tweaks)
- ▶ Misc collection of LADSPA plugins

Stuff not used in this session

- ▶ Lilypond (and \LaTeX) – typesetting and MIDI
- ▶ Hydrogen – Drum machine
- ▶ Timidity++ – Convert MIDI + Soundfonts to PCM audio

Alternatives

- ▶ Ardour – check it out if you want a pro-tools knock-off
- ▶ Audacity – easy to get started with
- ▶ Protux – also pro-tools wannabe?
- ▶ Others...



Starting JACK

- ▶ Set capture and playback ports (/proc/asound/pcm)
- ▶ Soft-realtime and small buffers for low-latency

```
hdparm -X66 -d1 -u1 -m16 -c3 /dev/hda
```

```
/opt/jack-bswap/bin/jackd -R -d alsa -C hw:1,1 -P hw:1,0 -p 64 -n 2
```

Concepts

- ▶ ecasound uses the concept of *chains* or *chainsetups*
- ▶ They're conceptually very similar to a patch lead
- ▶ They have two ends – an *input* and an *output*
- ▶ The difference is that you can add effects and junk

Chainsetup to record a track

- ▶ Chain 1 in from JACK and out to JACK
- ▶ Chain 2 in from JACK and out to track1.wav
- ▶ No effects. Software monitoring

```
ecasound -b:64 -c -a:1,2 -i jack_auto \  
-a:1 -o jack_auto -a:2 -o track1.wav
```

Chainsetup to record a second track

- ▶ Chain 1 in from JACK and out to JACK
- ▶ Chain 2 in from JACK and out to track2.wav
- ▶ Chain 3 in from track1.wav and out to JACK
- ▶ Each track in separate file

```
ecasound -b:64 -c -a:1,2 -i jack_auto -a:3 -i track1.wav \  
-a:1,3 -o jack_auto -a:2 -o track2.wav
```

Spot the pattern yet?

- ▶ Chain 1 in from JACK and out to JACK
- ▶ Chain 2 in from JACK and out to track3.wav
- ▶ Chain 3 in from track1.wav and out to JACK
- ▶ Chain 4 in from track2.wav and out to JACK
- ▶ Each track in separate file

```
ecasound -b:64 -c -a:1,2 -i jack_auto -a:3 -i track1.wav -a:4 -i track2.wav \  
-a:1,3,4 -o jack_auto -a:2 -o track3.wav
```

Listening to the results

- ▶ One chain for each file to play (input)
- ▶ Each chain to send output to JACK

```
ecasound -b:64 -a:1 -i track1.wav -a:2 -i track2.wav -a:3 -i track3.wav \  
-a:all -o jack_auto
```




Listening to the results – with effects

- ▶ Same as previous, but uses LADSPA Freeverb plugin on output
- ▶ Effect just attaches to a chain
- ▶ Could have added individual effects to individual chains

```
ecasound -b:64 -a:1 -i track1.wav -a:2 -i track2.wav -a:3 -i track3.wav \  
-a:all -eli:1123,0,0.5,0.4,0.1,0.5,0.4 -o jack_auto
```

Real-time effects

- ▶ To prevent latency problems, do as little in real-time as possible
- ▶ That includes resampling, byte-swapping and mixing software streams, yeah, I'm talking to you, hardware vendors!
- ▶ Sometimes real-time effects are necessary, *eg reverb on vocals*
- ▶ Many tools may be used for such processing...

Independent streams



ecasound for real-time effects

- ▶ Stereo stream from JACK read by two chains
- ▶ Split into two mono streams and send to separate chains
- ▶ Add effects to each chain
- ▶ Mix back into single stereo stream and give back to JACK

ecasound for real-time effects

```
ecasound -c -b:64 \  
-a:1,2 -i jack_auto \  
-a:1 -f:16,1,44100 -erc:1,1 \  
-eli:1123,0,0.5,0.4,0.1,0.5,0.4 \  
-o jack_generic,right \  
-a:2 -f:16,1,44100 -erc:2,1 \  
-eli:1123,0,0.5,0.2,0.1,0.5,0.5 \  
-o jack_generic,left
```



Connecting effects to JACK

```
jack_lsp # display JACK ports

jack_connect ecasound:left_2 coreaudio:Built-in\ Audio:in2 &&
jack_connect ecasound:right_1 coreaudio:Built-in\ Audio:in1
```

Processing individual tracks/takes

- Effects such as compression and reverb often applied here

```
ecasound -i mixr4.wav \  
-eca:30,0.2,0.50,0.50 \  
-eli:1123,0,0.5,0.2,0.1,0.5,0.5 \  
-o mixr4-e-compr.wav
```

Creating a master

- ▶ Effects may also be applied here to one or all tracks
- ▶ Panning and amplification often applied at this point
- ▶ Use `ecanormalize` on resulting mix

```
ecasound -a:1 -epp:50 -ea:90 -i vocal-take1r1.wav \  
-a:2 -epp:65 -ea:120 -i guitar-take1.wav \  
-a:3 -epp:35 -ea:120 -i guitar2-take2.wav \  
-a:all -o mixr1.wav
```


Cool new toys on their way

- ▶ Hydrogen improvements (including pitch)
- ▶ MIDI Reactor

Closing notes

Any questions?