An investigation of Inter-Application Audio Routing on the Macintosh OS X operating system.

Author: Richard Hallum

This document was presented as part of my final research for a Master of Music Technology Degree.

Table of Contents

1. Abstract 3
2. Aim 3
3. Objectives and outcomes 4
4. The Report 6
    - Background
    - The research process
    - Introduction
    - An overview of audio in OS X
    - The Problem
    - Third party inter-application audio routing
    - Miscellaneous audio routing software
    - Third party software routing solutions examined
    - Conclusion
5. Bibliography 16
6. Acknowledgements 22
7. Appendices 23

An investigation of Inter-Application Audio Routing on the Macintosh OS X operating system.

1 Abstract

The personal computer has now developed to the stage where it can be used to run several audio applications simultaneously. The Apple Macintosh OS X platform provides many advantages for music and multimedia users. It has an elegant and intuitive user interface which many creative users prefer. Pre-emptive multi-tasking, and protected memory provide this modern operation system with impressive speed and reliability[1]. Perhaps surprisingly, OS X cannot route audio between applications. This paper investigates why that is, and in particular examines latency and synchronisation issues that are inherently associated with audio streaming. These provide very real challenges for audio software developers who want to provide a solution to meet the demands of audio professionals. Several third party solutions do exist and these are examined, and compared.

2 Aim

The aim of this research is primarily to identify all available methods for routing audio between applications on the Macintosh OS X operating system. Any available software titles will be examined in practice to find one or more solutions and report on their effectiveness. It will also investigate what has been written to date on inter-application audio in OS X.

This project will:

- Review any literature on this subject
- Investigate how audio is supported on the Mac OS X operating system with the Apple technologies CoreAudio, and QuickTime
- Compare how MIDI is supported on the Mac OS X operating system with the Apple technologies CoreMIDI
- Compare how audio is supported on the Windows XP and Vista operating systems
- Evaluate third party software solutions for inter-application audio routing
- Evaluate other third party audio software solutions such as device selectors, and audio capture software
- Develop tutorial resources to instruct on the setup and use of some of the audio software tested
- Implement installation of the most suitable software solution on the MAINZ MIDI room computers

[1] This aspect of software design is examined in Appendix VIII

3 Objectives and Expected Outcomes

Objective	Outcome
To study CoreAudio (including QuickTime) to establish background information for this research project. This will involve sufficient depth to develop an understanding of Audio handling on OS X from a user perspective, and identify if inter-application audio routing is included in the OS X operating system. Detailed descriptions of programming will not be covered in this report.	A report on CoreAudio and QuickTime audio will be written A brief description on Windows audio will also be added. It will be established whether or not native support for inter-application audio routing is an inherent feature of OS X.
To survey all available third party software solutions for inter-application audio routing. A list of these is included in the resource section of this document. The intention is to trial the various software titles available, and where applicable conduct objective tests of increased latency and microprocessor usage. Combinations of certain third party audio routing and control solutions will also be included to identify possible conflicts.	Software examined will be Direct Connect, JACK, Rewire, and Soundflower. Test results for these will be discussed, tabulated, and graphed in the report.
To survey other available third party audio utility software for OS X. This will include any programme or system extension that can route or capture the audio in OS X. Other utilities (such as file converters, editing software) will not be considered in this report.	Software examined will be Audio Hijack, Detour, Line In, PTHVolume, Sound Menu, SoundSource, WireTap, and WireTap Anywhere. Test results for these will be provided in the report.
To trial the available software on a MacBook. This is the most common model of computer that is used by the Diploma in Audio Engineering & Music Production students at MAINZ. Therefore this particular model will be used for the most extensive part of the software evaluation.	Students will be able to install and use at least one of the audio routing solutions available. Instructions will be written, and screen-movie tutorials created.
To compare CoreAudio routing with Core MIDI routing. Since OS X version 10.3 there has been a solution for inter-application for MIDI data. It is therefore useful to relate the development of these two related software technologies.	A section on CoreMIDI will be included in the report.
To investigate the functionality of the new Aggregate Audio Devices feature that was added to the Audio MIDI Setup in OS X version 10.4	The Aggregate Audio Device will be used to create a full-duplex device on the Intel Mac.

To provide a written report of the findings of the software testing. This will detail all experiment results and provide a comparison of the features and effectiveness of each software programme. A glossary will be included. Background information on JACK will also be covered.	Results will be tabulated, and graphed where appropriate A detailed comparison chart of the evaluated software will provide a quick reference to the software attributes. A glossary of terms used in this report will be added. An in depth study of JACK will detail its theory of operation.
To accompany the report with QuickTime screen movies of any installation and/or setup procedures, where this method of explanation is preferable to using text and/or screenshots.	Setup procedures and screen movie tutorials will be created for using Soundflower, JACK, and Wiretap Anywhere.

4 The Report

4.1 *Background*

The Certificate in Audio Engineering and Music Production at MAINZ uses Windows computers for the MIDI / digital audio workstations. On this platform it is possible to route audio from various sources via the Mixer window. One example of use is to record the output of MIDI sounds (from the soundcard) back into an audio track of a sequencer programme. Students continuing on with the Diploma in Audio Engineering and Music Production use Macintosh computers and are often surprised to find that there is no similar audio mixer feature in OS X. A third part solution would therefore be highly desirable, particularly as the students have their own laptops.
A preliminary search on this topic indicated that few if any comprehensive surveys of software for inter-application audio routing have been undertaken. This report will therefore provide useful information for anyone using music/audio production software on a Macintosh computer.

4.2 *A Note on the Research Process*

This topic has been covered in MTEC6709 A1 and A2, and MTEC6710 A1 so only a summary is provided here. The original idea for this research was to find out what solution(s) exist to route audio between applications on the OS X operating system.
Initial research on the web indicated that several third party software solutions do exist, and these were duly tested. Altogether, a considerable number of possibly useful pieces of audio software were discovered. Not all of these can provide audio routing but they were evaluated in any case, and tested where appropriate.
To start with it appeared that very little information was available on this subject so I had to dig deep to find enough to work with. As time went on many resources were discovered and reviewed (as indicated by the bibliography), and this has enabled me to provide some background information on CoreAudio, and JACK. Some topic areas were infertile, eg no details of the operating principle of Soundflower were found.
Over the six-month period of this investigation I have stuck to the original direction of the research, i.e. answering the question ‘what is the best way to route audio between applications on OS X?’ The research was expanded somewhat to provide sufficient background information to enable the reader to understand the issues involved for the software developers who tackle this problem.
Originally, my thoughts were to test all the available Macintosh computer models but limit the testing to only JACK and Soundflower. Once testing was underway it became clear that to comprehensively test all combinations on all models (laptops, desktops, towers, Intel & PPC) as well as Tiger and Leopard would take more time than I had available so I scaled it back to three models, and two OS versions. Testing on the PPC Mac was also minimised, as this model is no longer used at MAINZ. In any case, testing has been rigorous and it is doubtful if more exhaustive testing would alter the outcome of this investigation.

4.3 *Introduction*

Over the last ten years personal computers have developed enormously in computing power. One of the areas that has benefited greatly from this advancement has been audio production. This has enabled a professional digital audio workstation (DAW) to be built using no more than a standard computer with suitable software and a hardware interface for audio I/O (input and output). In the 1980s computer music creation meant using the MIDI data format but in the mid to late 90s the ability to add audio recording and playback to MIDI sequencers became increasingly commonplace. Initially audio track counts were limited to two (i.e. stereo) but we have now reached the point where it is possible to have dozens of tracks recorded simultaneously, providing there is sufficient I/O, and many more bussed on the computer internally.

4.4 *An Overview of Audio in OS X*

Implementation of good audio handling was part of the original design of OS X. The previous operating system (OS 9) had reached its limits in many areas, including sound. Apple's system extension for sound (Sound Manager) was limited to 2-channel, 16 bit audio. There were also latency issues depending on what hardware was used (Wherry, 2003). To work around these limitations music software companies developed proprietary systems that address the OS and hardware at a lower level (Steinberg's ASIO, Emagic's EASI, MOTU's MAS, Digidesign's DAE).

Apple's solution in OS X is CoreAudio. This audio server (the hardware abstraction layer; HAL) sits between the hardware drivers and the application software. The benefit of this system is that programmers can develop audio applications without having to write for specific hardware, as the API's (Application Programming Interface) are addressing the HAL. This provides a consistent, high specification multi-channel audio routing system within OS X. Timing and synchronisation of audio signals are fundamental to the design, as is low latency (Wherry, 2003). Latency figures of 15ms were considered good on OS9. In OS X this can be reduced to around 1ms (Ray J., Ray W.C., 2003).

Closely associated with CoreAudio is Apple's QuickTime. QuickTime is best known as a media player application but beyond this visible part it also performs many other tasks, including media synchronisation and file conversion. A part of QuickTime is the MIDI implementation, which uses a DLS (Downloadable Sound Font) called the QuickTime Music Synthesizer. Audio output is handled by CoreAudio and will automatically be sent to the default audio device (Apple, 2005).

Also provided is a standard type of plugin architecture (Audio Units; AU) that any software manufacturer can use. Since Logic is an Apple product its plugins are entirely in the AU format. There are many freeware AU plugins now available. Other commercial developers have generally 'hedged their bets' by providing plugins in multiple formats. Digidesign continues to restrict Pro Tools to their own formats (AS, RTAS, TDM, HTDM).

For a more detailed explanation of CoreAudio and QuickTime refer to Appendix VII.

4.5 *The Problem*

The features of CoreAudio provide a stable and suitable way to handle most audio situations on OS X. Version 10.2 (Jaguar) introduced significant performance improvements, such as the Apple MIDI Setup Utility (Cooper, 2003) and since then several enhancements have been added, most notably the Aggregate Audio feature in Audio MIDI Setup which allows separate pieces of hardware to act as one virtual I/O.

With the introduction of Jaguar came unlimited multi-channel audio support and it is this feature that is most relevant to this study. The problem is that the design of multichannel functionality has been related to addressing the growing use of multichannel I/O. While this is natural and useful it has not kept pace with another development in audio software. Increasingly, audio production is being performed (sometimes completely) 'in the box'. Therefore, it is equally important for many users to be able to route 2-channel (and sometimes multi-channel) audio between applications within the same computer (Davis, 2006). Lack of support in this area is frustrating users (PeterB, 2008). Paul Davis says of his solution, JACK "much to my surprise it provides something Apple do not provide themselves, which is inter-application communication. I was under the impression that CoreAudio provided that, apparently it does not" (Davis, 2003).
OS X is rooted in UNIX, an OS where API's are based on the read/write model derived from the "everything is a file" abstraction. The problem with this design is that it fails to force developers to pay sufficient attention to the real-time nature of audio. In particular, it becomes difficult to facilitate inter-application audio routing if different programmes are not running synchronously (Davis, 2003).
To date only one 'roundup' of possible solutions has been found. Andi summarises six possible sound routing methods, and recommends Soundflower. His evaluation is somewhat limited. Although Jack is included he states "i didnt have the time to figure things out with jack os x" (sic) (Andi, 2007).

4.6 *Third Party Inter-Application Audio Routing*

A thorough search for utility software that can route audio between applications on OS X revealed two main possibilities (andi, 2007). These are Jack, and Soundflower, both of which are freeware (as are most of the utilities covered in this review).
Jack (Jack Audio Connection Kit) is an open source software solution and is written by Paul Davis and his associates. It is non-commercial in origin and to some extent relies on users having good computer literacy (Vucic, p23). Jack is accessed from two websites (www.jackaudio.org and www.jackosx.com) and does include a detailed manual. There is also a support forum on Yahoo (a Yahoo login is required). A network enabled audio communication tool (NetJack) is provided with Jack but this has been disabled in the latest version (0.77). "NetJack has been temporarily removed from the Jack OS X package, until it is fixed." (Jack OS X, p21). This indicates a weakness in this type of software in that it is still somewhat experimental compared to commercial release programmes. On the other hand, Vucic identifies that the software is conceptualised without having to be a commodity in a specific market and can therefore focus solely on the issue of inter-application data exchange (Vucic, p22). The design philosophy of Jack has been to provide a system to seamlessly move audio data between programmes and/or an audio interface (Davis, 2006). Although it is only on version 0.77, development has been quite fast, with 13 versions released to the public since its introduction in 2004. Version 0.75 introduced optimisation for running on dual processor machines (Jack OS X, 2008).
Jack's interconnections are realised using a virtual patchbay (the Connections Manager) to configure the list of inputs, outputs, and the connections between. One of Jack's key features is that it does not add any perceptible latency to the routed audio (Vucic, p25). Jack "has been designed from the ground up to be suitable for professional audio work." (Jack OS X, 2008). Synchronous execution of all clients (eg applications) is also a design priority. To achieve these goals Jack has a central audio engine called Jack Server that can communicate with all clients whether they are an I/O hardware interface, AU, or CoreAudio application.

Soundflower is also open source freeware and is provided by Cycling '74, primarily to accompany their product Max. It is a kernel driver and presents itself as a CoreAudio device (Ingalls, 2007). It is very effective for straightforward setups and can provide either 2 or 16 channels for audio routing, but is not as flexible as Jack (Davis, 2008, p43). The documentation is extremely brief

and while it is possible to search for Soundflower on the Cycling '74 website (www.cycling74.com) the information is biased toward Max support.
Thus there is a shortage of available support on this utility and the only way to find out its usefulness was to trial it. This can be an issue, as fine-tuning requirements for audio handling are common. Without installation, version and compatibility details any problems can easily go unresolved. One forum (PeterB, 2008) has two posts stating Soundflower is too buggy.
Both Jack and Soundflower do provide uninstallers, which is important as these utilities dig deeper into the operating system than most applications.

An alternative solution is Rewire. This is a commercial solution and is usually associated with Reason. Rewire can stream up to 256 audio channels as well as 255 MIDI busses between two audio applications (Propellerhead, 2008) and acts as a plugin. In a typical setup Reason is synchronised to a host (eg Logic, Pro Tools) and a MIDI track in the host can drive virtual synthesisers in Reason. Reason can in turn output audio back into tracks in the host application. Transport control can be from either the host or Reason. While many audio applications are now rewire enabled, the support is often to simply run a standalone version without the developer having to rewrite the programme source code as various plugins (AU, VST, RTAS). Rewire is written using the obsolete CFM (Code Fragment Manager) format and must be wrapped to work in a Mach-O (Mach Object file format), creating processing overhead (James, 2004). Functionality under Rewire is therefore varied, and it does not offer a universal approach to inter-application audio routing.

4.7 *Miscellaneous Audio Routing Software*

As well as utilities that route audio within the Macintosh computer there are several other utilities that enhance the ability of the user to select audio inputs and outputs (functions that are already handled with the Apple utilities 'Sound Preferences' and 'Audio MIDI Setup'). The third party offerings are small applications providing enhancements to the sound functionality. Line In (Rogue Amoeba, 2008) allows for passing the audio input directly through to the audio output. Sound Menu (Aspirine, 2007) and SoundSource (Rogue Amoeba, 2008) both allow menu-bar access to switch audio input and output. None of these solve the issue of inter-application audio routing. The convenience offered by having audio input and output switching from the menu-bar is important if used in conjunction with Jack or Soundflower as the audio source and/or destination will probably be set frequently.
An alternative approach to solving the problem is to capture the audio by recording it as it is streamed to the computer output. Two third party software solutions do this; WireTap allows you to record any audio playing through your Mac (Ambrosia Software). Audio Hijack performs the same task and also includes the ability to enhance the audio with effects plugins (Rogue Amoeba, 2008). A new version named WireTap Anywhere has now been released and is intended to supersede WireTap Pro. Interestingly, these programmes are the only commercial products available for OS X audio routing. These applications appear to provide a workable solution to the problem but fail to maintain a timing reference between applications. They do, however, offer the ability to capture audio from 'hidden' sources eg when a web browser handles streaming audio.
'Detour' is freeware form the Rogue Amoeba software company. It can "send different audio to different outputs, or lower the volume of some applications in relation to others" (Rogue Amoeba, 2005). This used to be a commercial utility and is no longer supported[2] but is still available and suitable for PPC (PowerPC) Macintosh machines running Tiger (OS X 10.4).
Evaluation of these and other audio software utilities are in appendix IV.

[2] the final release was version 1.5.5 dated Oct. 2005 (Detour readme.pdf)

4.8 *Third Party Software Routing Solutions Examined*

a. Soundflower

i. Soundflower v1.2.1

Soundflower was developed by Cycling 74 and is open source software. It is a background application in the form of a kernel extension and has no GUI. It is accompanied by the optional Soundflowerbed, which adds a menu to the menu bar with the following dropdown menu items:

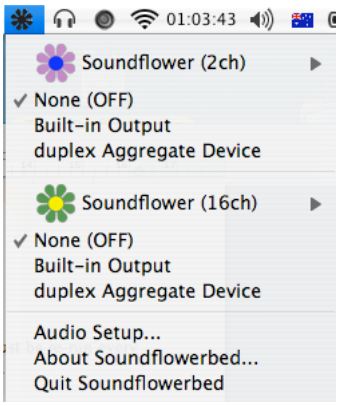

Fig. 1 Soundflowerbed menu items (Cycling '74)

Installation of Soundflower was straightforward. Soundflowerbed does not need installation as the programme runs when the icon is double-clicked, and shows a flower icon in the menu bar. Soundflowerbed must be rerun every time the machine is booted, unless installed as a startup item [system prefs/accounts]. If Soundflowerbed is run without Soundflower, a menu message appears in place of the Soundflowerbed menu items: "Soundflower is not installed!!" Uninstalling requires using the Terminal to run a shell script. Soundflowerbed tends to hang when the uninstaller runs. Logging out and in again fixed this. It is therefore best to not have Soundflowerbed running during the uninstall process.

To use Soundflower it is not necessary to use Soundflowerbed, as it is just a matter of selecting Soundflower in the input and output audio applications. Some audio applications do not access to inputs and/or outputs. In this case Soundflower must be selected in either the Sound pane (System Preferences), or in Audio Devices (Audio MIDI Setup). A handy menu item in Soundflowerbed is Audio Setup… which opens the Audio MIDI Setup. It is also possible to set the input and output connections from the Soundflowerbed menu. This can be done for 2 channels or 16 channels (although few applications can provide 16 channel access). An example is shown below, where the Built-in Output has been assigned to Soundflower.

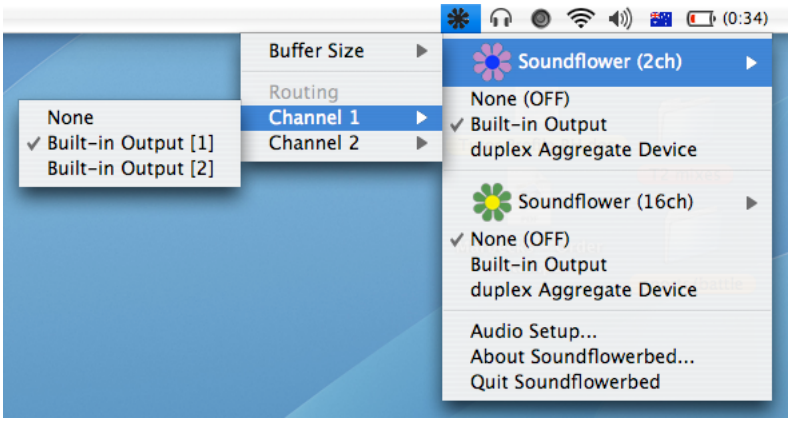

Fig. 2 Selecting outputs in Soundflowerbed (Cycling '74)

Buffer size is adjustable from 64 to 2048 samples and is very tolerant on a MacBook. Any setting from 64 to 2048 samples could be used with no sign of glitching. Latency is set by the Logic buffer setting, and adjusting the Soundflower buffer had no effect. Native Intel Mac support was introduced in version 1.2.

ii. Soundflower version 1.3.1

This version has been modified to allow use of volume, mute, and gain controls in the AMS, and balance control in the Sound preference pane. It is a patch of version1.3, which was an unofficial release bundled with some Rogue Amoeba software, and is currently maintained by Joachim Bengtsson (http://thirdcog.eu). Installation and performance on an Intel iMac was identical to version 1.2.1 with no problems arising during testing.

b. JACK

i. JACK version 0.77

Installation of JACK is reasonably straightforward. A full listing of exactly where each of the forty files goes is included in the documentation. An uninstaller is provided. This runs an uninstall command from the Terminal and logs the uninstall activity. On the MacBook the Jack Router appears in the AMS Audio Devices menus but did not always show in the Sound preference pane. Setup (as distinct from installation) of JACK is more involved than Soundflower. Several things must be set before the router can be used. On an Intel Mac the Sound in and Sound out are ½ duplex. JACK treats all clients as mono full duplex and cannot deal with ½ duplex, so they must be combined in an AMS aggregate device for JACK to be able to communicate with them. The aggregate device must be created using an administrator account. JACK is also unable to process stereo interleaved audio, but this is not so much of an issue as stereo tracks are de-interleaved on playback from the client application. Secondly, correct setup order is important. Details of the correct setup procedure are shown in Appendix III of this report. If a strict setup sequence is not followed applications will fail to appear in the Connections Manager. In any case some applications will not show in the Connections Manager until they are actually playing an audio file. This is identified in the manual but is unusual software behaviour and could cause users some confusion.

The Connections Manager interface is somewhat ambiguous in operation. Double clicking is performed to make a connection. This can be done on either the Send or Receive device (so long as the Receive device is highlighted). Some form of graphical patchbay would greatly enhance the user experience.

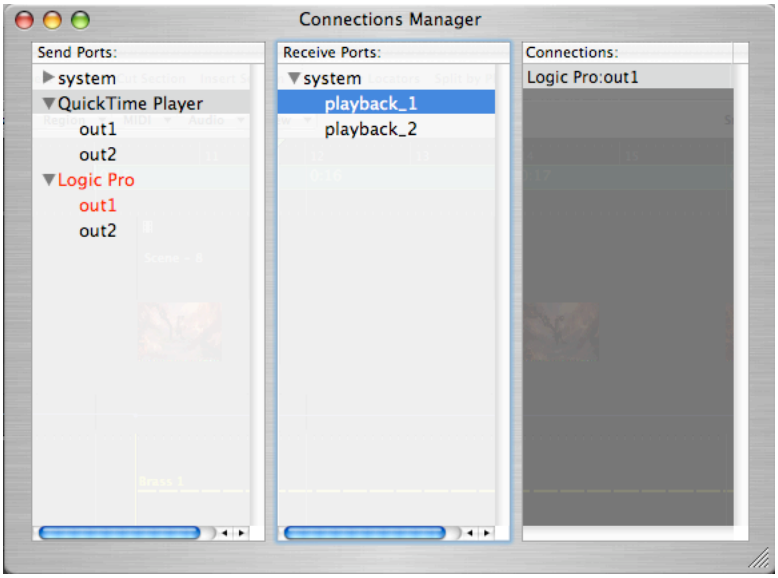

Fig. 3 The Connections Manager showing Logic output going to the Built-in Sound (Davis P., 2008)

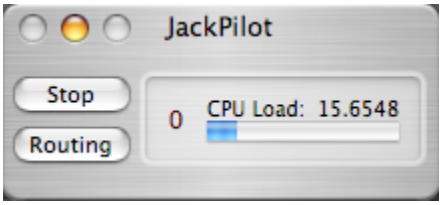

Fig. 5 The JackPilot Window, showing CPU load on a MacBook (Davis P., 2008)

JackPilot is the simple GUI for JACK and provides a button to start/stop the JACK server. The Routing button opens the Connections Manager window. CPU load shown in JackPilot. The value shown is sometimes considerably lower compared with that shown in the Apple Activity Monitor utility. Where applicable, both results have been included in Appendix I of this report.

For the tests the Connections Manager allowed two possible configurations: i) connecting from the Logic output into the System audio and then connecting the system audio output into the Logic input, or ii) connecting the Logic output to its input directly. This is the preferred option as the levels remain at unity gain.

For the latency tests JACK had to be persuaded to allow virtual loopback from Logic Audio output to input. This was done by setting a parallel input into Logic from QuickTime. For some reason Jack will not reconnect Logic's output until something else is connected to its input (the Logic output continues to default to the built in audio output). Once a connection is made from QuickTime to Logic's input then Logic's output can be seen on a record enabled channel in Logic. If the QuickTime connection is deleted the Logic connection also disappears. Feedback is prevented as JACK allocates separate busses for each connection. JACK version 0.78 was also tested (on Leopard, 10.5.4), and does allow loopback of Logic output to input audio, and therefore did not require this workaround.

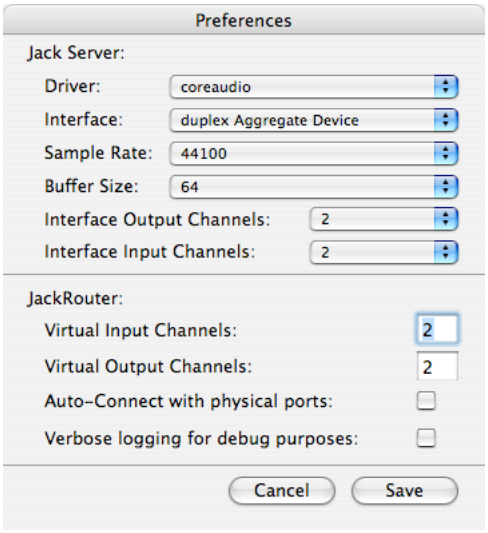

Fig. 6 The JACK Pilot Preferences window (Davis P., 2008)

Preferences can be set for a number of JACK parameters. Buffer sizes range from 32 to 2048 samples, and the Sample Rate can be set to 44.1kHz, 48kHz, or 96kHz. During latency testing, the latency was set by the JACK buffer setting. Adjustment of the Logic buffer size has no effect on the latency results. This is the opposite behaviour to Soundflower, where Logic overrode the router buffer size setting. Two special situations where the JACK buffer must be set to 1024 are included in the documentation. These are when using the JACK AU plugin or the Apple DVD Player (Davis, 2008).

The number of input and output channels is also set in the preference window, for both the audio interface and the virtual channels that will be used for inter-application communication. JACK also provides for saving Connections Manager setups (these are called Studio Setups) and this is a real advantage for complex or seldom used configurations.

MIDI is supported in JACK's architecture, but not implemented. At the time of writing there is no intention on the developers part to go ahead with MIDI on OS X, as inter-application MIDI communication is now available using the IAC busses.

If the JACK server is quit without first shutting down the clients then Logic remains stable and senses the loss of the JACK device then switches to the default device (built-in audio). QuickTime handles this situation less favourably and needs to be quit. If the JACK server is restarted at this point it does not respond and so no connections are made. It is therefore a good idea to heed the warning message and not stop the JACK server while clients are running. One strange attribute of JACK is that JACK Pilot can be quit without shutting down the JACK server. It is uncertain whether this was a design feature or simply occurs as a result of the modularity of the three part JACK system.

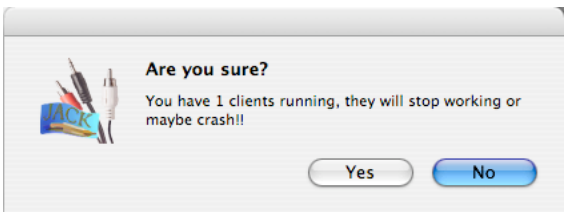

Fig. 7 JACK warning message (Davis P., 2008)

Details of JACK's principles of operation are in Appendix V. Basic testing of QJackCtl, an alternative GUI for JACK, was also undertaken. For more information refer to Appendix V.

c. Rewire

i. Rewire v1.7

Rewire is another way of getting audio transferred between two applications. It is a product of Propellerhead Software, and is primarily used with their virtual rack / sequencer Reason, although a number of other software products are compatible with Rewire. Most commercial sequencers can act as host (Logic, Cakewalk, Digital Performer, Pro Tools, Cubase). Rewire appears as a plugin and allows audio and MIDI data to be transferred between the host and any rewire enabled application.

According to the limited information available on the website, Rewire offers high-precision synchronisation and complete glitch-free sync (www.propellerhead.se, 2008). Rewire can support a total of 256 audio channels. These may be shared across several clients (Reason has a maximum of 64 channels). It can also transmit MIDI signals between applications with a maximum of 4080 MIDI channels (255 x 16). An additional feature is that, where applicable, the two applications can have their transports synchronised. It is unclear if these claims relate to this, or to sample-accurate synchronisation of the streamed audio. Rewire is a reliable piece of software but did show some instability with MIDI routing (see Appendix X). The audio busses can be disabled and enabled during audio streaming with no problems. Latency has been specified as 64 samples (1.45 mS) (Walker, 1999). CPU usage for Rewire itself is probably insignificant but the combined usage of Pro Tools and Reason being connected via Rewire is about 45% on a MacBook.

Rewire is easy to setup, as there are no settings on the plugin. The sample rate is set at 44.1kHz and the buffer size setting is automatic. It does offer reliable interchange of audio and MIDI between specific host and client applications, with sample-accurate synchronisation. The transport synchronisation is also reliable and the two programmes run in parallel so the transport functions of either will work equally well.

Fig. 8 The Rewire setup. Audio is being routed from Reason to Pro Tools and MIDI is being routed from Pro Tools to Reason

d. DirectConnect

In 2000 Digidesign introduced a similar system of audio interconnection software called DirectConnect. It was a TDM or RTAS plugin allowing streaming of up to 32 channels of 24-bit audio from an audio application into Pro Tools. It is no longer available.

4.9 *Conclusion*

This research has set out to provide a comprehensive survey of audio routing on the OS X operating system, evaluation and testing of the audio utility software, and creation of tutorials on its use.
All of the software tested performed 'remarkably well'[3], with very few actual crashes. Furthermore most of the software tested was tolerant of other audio utilities running concurrently. Much of this robustness can be attributed to the exceptional stability of OS X. CoreAudio offers an effective, low latency, sample accurate audio architecture for OS X. By design, it does not, however, feature inter-application audio routing.
Several possible audio routing solutions do exist but some are specialised (eg the DLSMusicDevice), or are limited to certain applications (eg Rewire). Also some must work as plugins (eg AUNetSend / Rec). Only JACK and Soundflower offer a global solution.
JACK is a professional non host-based solution, which adds zero latency to synchronised audio. Although it is an elegant piece of software, JACK's architecture is somewhat complex and this requires more care with installation and use. JACK can use an appreciable amount of the CPU (eg 30% on a PPC Mac, 20% on a MacBook), and this increased with higher sample rates (eg 57% @ 96kHz). Applications that don't allow the user to separately choose input and output drivers can be handled better by JACK than Soundflower. Also JACK's AU plugin provides for more complex patching configurations.
Soundflower is a simpler solution, yet surprisingly robust. In contrast to some of the software tested, it appeared in every audio device list. Soundflower's best latency was 2.9mS and while not matching JACK (0.73mS) it is a very small amount. Due to its ease of installation, setup, and use I would recommend it for education use. It will be installed on all the MAINZ MIDI computers, and suggested as the preferred solution for students to use on their laptops.
A number of other audio routing and control utilities were evaluated, and in some cases audio capture software (eg Audio Hijack or WireTap Anywhere) will be the best solution when stereo audio needs to be detoured into a file. Additional utilities such as Sound Menu, SoundSource and PTHVolume provide a handy way to select devices and control volume.
It should be noted that for many users transferring audio data by way of file exporting/ importing will be the preferred method to get audio between applications as many tracks of audio can be exchanged faster than in real-time. If time-stamped formats (such as broadcast wavefiles, AAF) are used then synchronisation of the audio is maintained.
Setup procedures for three common scenarios have been documented in Appendix III and accompanying video tutorials produced (on CD2).

[3] Mac users who have migrated from OS9 to OS X will best appreciate this statement

5 Bibliography

ALSA Project, 2008, Advanced Linux Sound Architecture (ALSA) project homepage, *ALSA Project*, accessed 17.9.08
http://www.alsa-project.org/main/index.php/Main_Page

Andi, 2007, Sound Routing, *Sojamo*, accessed 12.6.08
http://www.sojamo.de/blog/2007/10/15/sound-routing/

Apple Inc., 2008, Audio*, Apple Developer Connection,* accessed 24.5.08
http://developer.apple.com/audio/

Apple Inc., 2008, Audio Unit Programming Guide, *Apple Developer Connection,* accessed 24.5.08
http://developer.apple.com/documentation/MusicAudio/Conceptual/AudioUnitProgrammingGuide/AudioUnitProgrammingGuide.pdf

Apple Inc., 2006, Architecture of Mac OS X Audio, *Apple Developer Connection,* accessed 28.5.08
http://developer.apple.com/documentation/DeviceDrivers/Conceptual/WritingAudioDrivers/AudioOnMacOSX/chapter_2_section_3.html

Apple Inc., 2004, CoreAudio-Introduction, *Apple Developer Connection,* accessed 28.5.08
http://developer.apple.com/documentation/MusicAudio/Reference/CoreAudio

Apple Inc., 2007, CoreAudio Technologies, *Apple Developer Connection,* accessed 28.5.08
http://developer.apple.com/audio/coreaudio.html

Apple Inc., 2007, Introduction to Mac OS X Technology Overview, *Apple Developer Connection*, accessed 6.9.08
http://developer.apple.com/documentation/MacOSX/Conceptual/OSX_Technology_Overview/About/chapter_1_section_1.html#//apple_ref/doc/uid/TP40001067-CH204-TPXREF101

Apple Inc., 2006, Introduction to QuickTime Musical Architecture, *Apple Developer Connection,* accessed 24.5.08
http://developer.apple.com/documentation/QuickTime/RM/MusicAndAudio/qtma/A-Intro/chapter_1000_section_1.html

Apple Inc., 2008, Mac OS X Snow Leopard, *Apple,* accessed 20.6.08
http://www.apple.com/macosx/snowleopard/

Apple Inc., 2006, Technical Note TN2091-Device input using the HAL Output Audio Unit, *Apple Developer Connection,* accessed 16.6.08
http://developer.apple.com/technotes/tn2002/tn2091.html

Apple Inc., 2005, QuickTime Overview-Architecture, *Apple Developer Connection,* accessed 24.5.08
http://developer.apple.com/documentation/QuickTime/RM/Fundamentals/QTOverview/QTOverview_Document/chapter_1000_section_2.html

AppleInsider, 2004, CoreAudio in Mac OS X Tiger to improve audio handling, *Apple Insider,* accessed 29.5.08
http://forums.appleinsider.com/showthread.php?s=&threadid=45567

Ardour Foundation, 2007, Getting Audio In, Out and Around Your Computer, *Ardour Manual*, accessed 6.6.08
http://ardour.org/files/manual/sn-conFig.uring-jack.html

Ash M., 2006, Why CoreAudio is Hard, mikeash.com, accessed 6.6.08
http://www.mikeash.com/?page=pyblog/why-coreaudio-is-hard.html

Audio Engineering Society, 2006, AES information document for digital audio – Personal computer audio quality measurements (AES-6id-2006), *AES*, accessed 15.9.08
www.aes.org/publications/standards/preview.cfm?ID=6

Bengtsson J., 2008, No Sound Plays on PPC Macs, *Get Satisfaction*, accessed 17.10.08
http://getsatisfaction.com/cycling74/topics/1_3_no_sound_plays_on_ppc_macs_works_fine_on_intel

Cakewalk, 2008, Latency: What's Required vs What's Possible, *Cakewalk DevXchange*, accessed 18.9.08
http://www.cakewalk.com/DevXchange/audio_i.asp

Chartier D., 2008, SoundSource 2: Real audio controls in your menu bar, *Ars Technica*, accessed 7.9.08
http://arstechnica.com/journals/apple.ars/2008/03/04/soundsource-2-real-audio-controls-in-your-menu-bar

Cohen P., 2008, WireTap Anywhere lets you redirect Mac audio, *Mac Publishing*, accessed 19.9.08
http://www.macworld.com/article/135011/2008/08/wta.html

Cooper S., 2003, Audio and MIDI under Mac OS X Revisited, *NZMac.com*, accessed 20.5.08
http://www.nzmac.com/features/multimedia/audio-and-midi-under-mac-os-x-revisited.html

Corbett R.,van den Doel K., Pai D., 2002, Evaluation of Low Latency Audio Synthesis using a Native Java – ASIO Interface, *Dept. of Computer Science, University of British Colombia*, accessed 9.11.08
http://www.cs.ubc.ca/~rcorbett/lat02.pdf

Cosgrovee K. 2005, [linux-audio-user] audacity jack, *linux-audio-user*, accessed 5.9.08
http://music.columbia.edu/pipermail/linux-audio-user/2005-July/024519.html

Davis P., 2003, The JACK Audio Connection Kit, *Linux Audio Systems*, accessed 10.9.08
http://jackaudio.org/documentation

Davis P., 2006, Requirements for OS X, *Ardour*, accessed 13.9.08
http://ardour.org/osx_system_requirements

Dominic, 2008, WireTap Anywhere progress log, *Ambrosia Software Inc*., accessed 24.6.08
http://www.ambrosiasw.com/forums/index.php?showtopic=119722

Donner M., 2006, Making Connections with Rewire, *Penton Media Inc*., accessed 15.10.08
http://emusician.com/mag/emusic_making_connections_rewire/

Ekeroot J., 2007, Audio Software Development – an Audio Quality Perspective, *Luleå University of Technology*, accessed 14.9.08
epubl.ltu.se/1402-1552/2008/059/LTU-DUPP-08059-SE.pdf

Epson, 2001, Inside Mac OS X: System Overview, *ZDnet.co.uk*, accessed 13.6.08
http://whitepapers.zdnet.co.uk/0,1000000651,260014354p-39000512q,00.htm

Fielding S., 2008, QjackCtl and the Patchbay, *rncbc.org aka Rui Nuno Capela*, accessed 9.10.08
http://www.rncbc.org/drupal/node/76

Fiera D., 2008, WireTap Anywhere progress log, *Ambrosia Software Web Board*, accessed 11.9.08
http://www.ambrosiasw.com/forums/index.php?showtopic=119722

Figlar N., 2007, Jack Quickstart Guide, *64 Studio*, accessed 9.10.08
http://64studio.com/quickstart_jack

Fischmann S., 2006, Free OS X Audio Utilities, *Jelsoft Enterprises Ltd*, accessed 22.9.08
http://www.soundsonline-forums.com/archive/index.php/t-5012.html

Gore J., nd, How to Maximize Your Inputs with Aggregate Audio, *Apple Pro Techniques*, accessed 2.6.08
http://www.apple.com/pro/techniques/aggregateaudio/

Haddad P., 2008, PTHVolume 2, *PTH Consulting*, accessed 8.9.08
http://pth.com/products/pthvolume/

Isaacson C., 2007, Software Pipelines, *Quovadx Inc*., accessed 24.9.08
http://www.roguewave.com

James D., 2004, Linux Jack Sound Server Ported to OS X, *Sound on Sound*, accessed 12.9.08
http://www.soundonsound.com/sos/Oct04/articles/applenotes.htm

James J. D., 2007, Internal Mac Sound Recorder?, *Ask MetaFilter*, accessed 22.5.08
http://ask.metafilter.com/66436/Internal-Mac-Sound-Recorder

Kerner S., 2004, Open Source Awards 2004: Paul Davis for JACK, *CNet Networks*, accessed 9.10.08
http://articles.techrepublic.com.com/5100-10878_11-5136755.html

Kirn P., 2008, Leopard Audio Woes and Digidesign; 10.5.2 is a lemon for music?, *Create Digital Media*, accessed 17.10.08
http://createdigitalmusic.com/2008/05/21/leopard-audio-woes-and-digidesign-1052-is-a-lemon-for-music/

Kirn P., 2007, Leopard: Incompatibilities with JACK, Soundflower, *Create Digital Media*, accessed 17.10.08
http://createdigitalmusic.com/2007/12/03/leopard-incompatibilities-with-jack-soundflower-finder-audio-previews/

Kirn P., 2007, Vista for Music + Pro Audio, *Create Digital Music*, accessed 18.9.08
http://createdigitalmusic.com/2007/01/19/vista-for-music-pro-audio-exclusive-under-the-hood-with-cakewalks-cto/

Kuper R., 2004, Multiprocessing in SONAR 3.1, *Cakewalk DevXchange*, accessed 18.9.08
http://www.cakewalk.com/DevXchange/multiprocessing.asp

Lei Lei, 2007, iVol by Livecn, *www.huasing.org,* accessed 10.9.08
http://livecn.huasing.org/ivol/

Letz S., Orlarey Y., Fober D., 2005, Jack audio server for multiple machines, *Grame Research*, accessed 10.9.08
http://www.grame.fr/Recherche/Publications/list/index.php?p=list.php?lang=uk&type=ARCH

Letz S., Orlarey Y., Fober D., 2005, jackdmp: Jack server for multi-processor machines, *Grame Research*, accessed 10.9.08
http://www.grame.fr/Recherche/Publications/list/index.php?p=list.php?lang=uk&type=ARCH

Letz S., Orlarey Y., Fober D., Davis P., 2004, Jack Audio Server: MacOSX port and multi-processor version, *Grame Research*, accessed 10.9.08
http://www.grame.fr/Recherche/Publications/list/index.php?p=list.php?lang=uk&type=ARCH

MacMillan K., Droettboom M., Fujinaga I., 2001, Audio Latency Measurements of Desktop Operating Systems, *Peabody Institute of the John Hopkins University*, accessed 4.11.08
http://www.music.mcgill.ca/~ich/research/icmc01/latency-icmc2001.pdf

MacMusic, nd, Music & Audio on Macintosh, *MacMusic*, accessed 18.6.08
http://www.macmusic.org/home/?lang=en

McIntosh J., Toporek C., Stone C., 2003, Mac OS X in a Nutshell, *O'Reilly and Associates*, Sebastopol, CA, USA.

Mertin O., 2004, A Big Mac and a Side of Plug-Ins, *Penton Media*, accessed 16.10.08
http://emusician.com/mag/emusic_big_mac_side/

moki, 2008, Ambrosia seeks WireTap Anywhere (MacOS X) beta test, *BigBlueLounge.com*, accessed 24.6.08
http://www.bigbluelounge.com/forums/viewtopic.php?t=43395&sid=2c83c56afc0741a8ee25ec1771dc4cfc

Moore J., 2006, Re: WireTap, CoreAudio's API, and system capture, and kexts…, *Apple Mailing Lists*, accessed 27.10.08
http://lists.apple.com/archives/coreaudio-api/2006/Jan/msg00101.html

Moore M., 2006, *Apple Mailing Lists*, accessed 27.10.08
http://lists.apple.com/archives/coreaudio-api/2006/Jan/msg00101.html

Native Instruments, 2008, Mac OS X Tuning Tutorial, *Native Instruments Support*, accessed 20.9.08
http://www.native-instruments.com/index.php?id=niosxtut5&L=1

Native Instruments, 2007, Native Instruments Setup Guide, *Native Instruments*, Berlin, Germany

Orenstein D., 2000, Quickstudy: Application Programming Interface (API), *Computerworld Inc*.
http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=43487

Parcher M., 2005, An alternative method for recording computer audio, *Mac OS X Hints*, accessed 8.6.08
http://www.macosxhints.com/article.php?story=20051219161639252

PeterB, 2008, Any alternative to Soundflower to route audio output into input?, *ArsTechnica Openforum*, accessed 12.6.08
http://episteme.arstechnica.com/eve/forums/a/tpc/f/8300945231/m/746003552931

Phillips D., 2007, Jack Sync: A Primer for Linux Users, *Linux Journal*, accessed 9.10.08
http://www.linuxjournal.com/node/1004080

Pitcarn J., 2007, Record offset in Logic, *Opus Locus*, accessed 4.11.08
http://www.opuslocus.com/logic/record_offset.php

Poole L., Cohen D., 2002, Macworld Mac OS X Bible, *Hungry Minds*, NY, USA

Propellerhead, 2008, Rewire, *Reason*, accessed 27.6.08
http://www.propellerheads.se/products/reason/index.cfm?fuseaction=get_article&article=rewire

Quentin, 2005, New Audio Units in OS X 10.4, *Rogue Amoeba Software*, accessed 23.9.08
http://www.rogueamoeba.com/utm/posts/Article/autiger-2005-05-19-21-00

Ray J., Ray W.C., 2003, Mac OS X Unleashed, 2nd ed.,Sams, Indianapolis, IN, USA

Rudolph B., 2003, Steinberg VST System Link, *Mix Online*, accessed 21.9.08
www.barryrudolph.com/mix/pdfs/steinbergvst.pdf

Schotman H., 2005, Basic Studio Setup using Audio Hijack Pro (v2.0), *Hugo Schotman, Zurich*, accessed 15.10.08
http://log.hugoschotman.com/hugo/2005/02/basic_studio_se.html

Schotman H., 2005, How to Add or Change a Soundflower Device, *Hugo Schotman, Zurich*, accessed 17.10.08
http://log.hugoschotman.com/hugo/2005/04/how_to_add_or_c.html

Schotman H., 2005, Soundflowerbed & Skype don't seem to like each other, *Hugo Schotman, Zurich*, accessed 16.10.08
http://log.hugoschotman.com/hugo/2005/03/soundflowerbed_.html

Sellers D., 2004, Mac OS and the music plug-in situation, *Macsimum News*, accessed 2.6.08
http://www.macsimumnews.com/index.php/archive/mac_os_and_the_music_plug_in_situation

Shaffer J., Rosenzweig G., 2004, MacAddict guide to Making Music with GARAGEBAND, *Que*, CA, USA

Shirkey P., 2006, JACK user documentation, *Linux Audio Users Guide*, accessed 16.9.08
http://lau.linuxaudio.org/jack/

Soundforge, 2008, Midishare, *Soundforge.net,* accessed 25.5.08
http://sourceforge.net/projects/midishare/

Soundforge, 2008, Jack OS X, *Soundforge.net,* accessed 25.5.08
http://sourceforge.net/projects/jackosx

Tim, 2007, Getting to Know JACK (QjackCtl), *64 Studio*, accessed 9.10.08
http://www.64studio.com/manual/audio/jack

Ubuntu documentation, 2007, HowToQjackCtlConnections, *Canonical Ltd*., accessed 9.10.08
https://help.ubuntu.com/community/ HowToQjackCtlConnections

Vucic V., nd, Free Software Audio Applications for Audio Playback, Recording, Editing, Production and Radio Broadcast Management and Automation, *Linux Center, Serbia and Montenegro*, accessed 17.5.08
http://www.gnulinuxcentar.org/Audio_Tools_Scan.pdf

Walker M., 1999, Rewired for Sound, *Sound on Sound Ltd*, accessed 15.10.08
http://www.soundonsound.com/sos/nov99/articles/rewire.htm

Wherry M., 2003, Mac OS X For Musicians, *Sound on Sound*, accessed 27.5.08
http://www.soundonsound.com/sos/Apr03/articles/osx.asp?print=yes

Wherry M., 2005, Mac OS X Tiger: A Musician's Guide, *Sound on Sound*, accessed 27.5.08
http://www.soundonsound.com/sos/jul05/articles/tiger.htm

Wiffen P., 2004, Investigating CoreAudio Performance Under Mac OS X, *Sound on Sound*, accessed 27.5.08
http://www.soundonsound.com/sos/nov04/articles/coreaudio.htm

Wikipedia, 2008, Pipeline (software), *Wikimedia Foundation*, accessed 12.9.08
http://en.wikipedia.org/wiki/Pipes_and_filters

Winkler P., Shirkey P., Rzewnicki E., Knecht M., Low-Latency HOWTO, *linuxaudio.org*, accessed 9.10.08
http://lowlatency.linuxaudio.org/

6 Acknowledgements

I would like to thank Heather for allowing me to skip doing dishes whenever an assignment was due in.