Panoramix

Quick Start Manual

IRCAM – CNRS UMR 9912 STMS
1 place Igor-Stravinksy,
75004, Paris, France
http://www.ircam.fr

Last update: March 10, 2017, Thibaut Carpentier

Contents

1 Introduction

panoramix is a post-production workstation for 3D-audio contents. This tool offers a comprehensive environment for mixing, reverberating, and spatializing sound materials from different microphone systems: surround microphone trees, spot microphones, ambient miking, Higher Order Ambisonics capture. Several 3D spatialization techniques (VBAP, HOA, binaural) can be combined and mixed simultaneously in different formats. *panoramix* also provides conventional features of mixing engines (equalizer, compressor/expander, grouping parameters, routing of input/output signals, etc.), and it can be controlled entirely via the Open Sound Control protocol (https://en.wikipedia.org/wiki/Open_Sound_Control).

It is assumed that the reader of this manual is somehow familiar with audio engineering and sound spatialization techniques.

2 Requirements and Installation

2.1 System Requirements

panoramix is compatible with:

- macOS X (Intel processor) **10.7** or higher.
- Windows 7 or higher.
- Linux x64 with the JACK Audio Connection Kit.

2.1.1 macOS requirements

For optimal performances, *panoramix* makes use of Streaming SIMD Extensions (SSE)(see e.g. http://en.wikipedia.org/wiki/Streaming_SIMD_Extensions).
Various SIMD Extensions exist and your processor may or may not support all of them.
You can check the extensions supported by your hardware by typing the following command in a Terminal: "sysctl -a | grep hw.optional" .
(a value of "1" indicates that the extension is supported; "0" is not supported).
panoramix requires the following extensions: SSE, SSE2, SSE3, SSE3 supplemental and SSE 4.1.

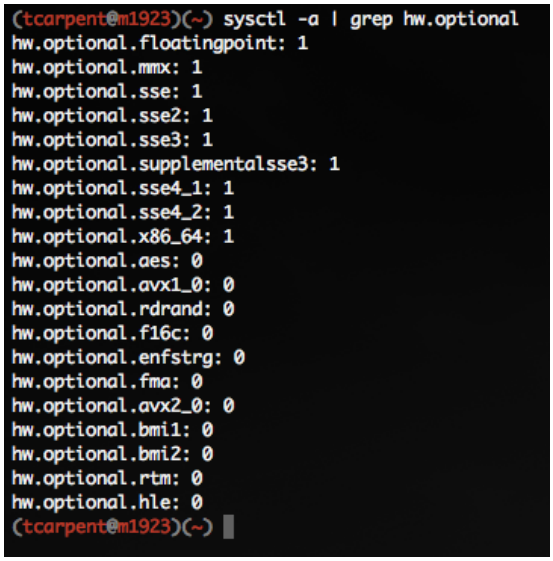

Figure 1: Checking SIMD extensions

2.1.2 Windows requirements

panoramix Windows version requires the "Visual C++ Redistributable Packages for Visual Studio 2015". You can (freely) download it from Microsoft website: https://www.microsoft.com/en-us/download/details.aspx?id=48145.
Select either "vcredist_x86.exe" or "vcredist_x64.exe" depending on your architecture, and follow the instructions for installation.

2.2 Installation

- mount the *panoramix* disk image (i.e. double-click on the *panoramix* .dmg file).
- copy the *panoramix* application somewhere on your computer (you can simply drag-and-drop the *panoramix* icon to the desired folder).
- plug and configure your audio sound-card.
- double-click the application.

3 Getting started

panoramix is a software mixing console that can process up to 128 input and 64 output channels. A multichannel audio driver is therefore required. *panoramix* receives audio signals from the input audio driver, processes them, and then delivers audio signals to the output driver. Either a hardware audio sound-card or a virtual audio driver can be used. *panoramix* supports any CoreAudio compatible driver.

On macOS, several virtual audio drivers are available, e.g. JACK Audio Connection Kit (http://www.jackaudio.org), Soundflower (https://cycling74.com/soundflower), Loopback (https://www.rogueamoeba.com/loopback), etc.
On macOS, it is further possible to combine both hardware device and virtual driver, by creating a so-called "aggregate audio device". See Apple documentation for further informations.

See Fig. 2 for an overview of the hardware and virtual driver workflows.

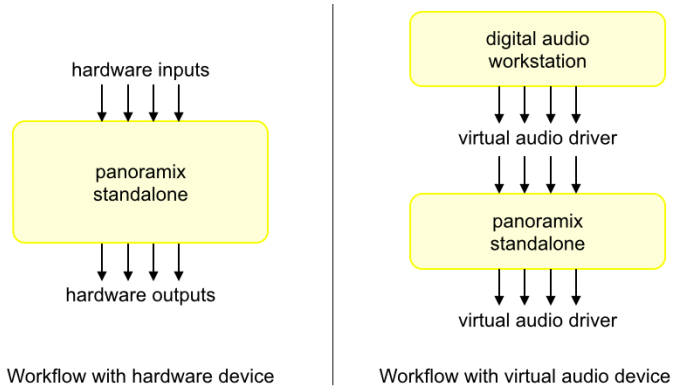

Figure 2: Possible workflows for operating *panoramix*

When you launch the *panoramix* application, two windows will pop up: the "application window" (Fig. 3) and the "console window" (Fig. 4).

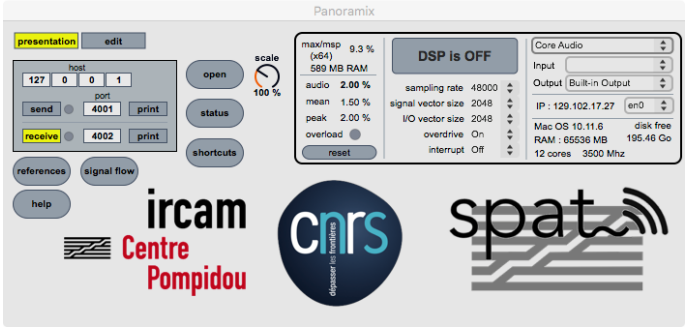

Figure 3: *panoramix* application window

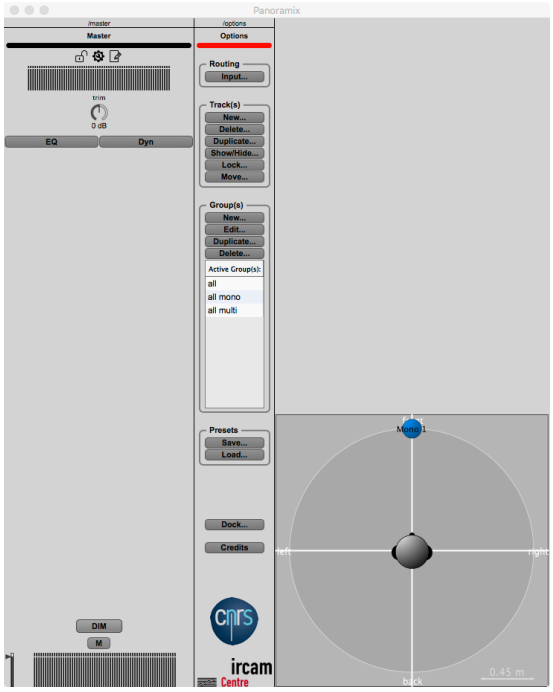

Figure 4: *panoramix* console window

The "application window" is used to configure the *panoramix* application, prior to any audio rendering. The "console window" is the place where all the sound mixing will occur.
First, go to the "application window" (Fig. 5), select your audio driver, choose the appropriate input and output audio devices, select the sampling rate, and switch on the audio processing.

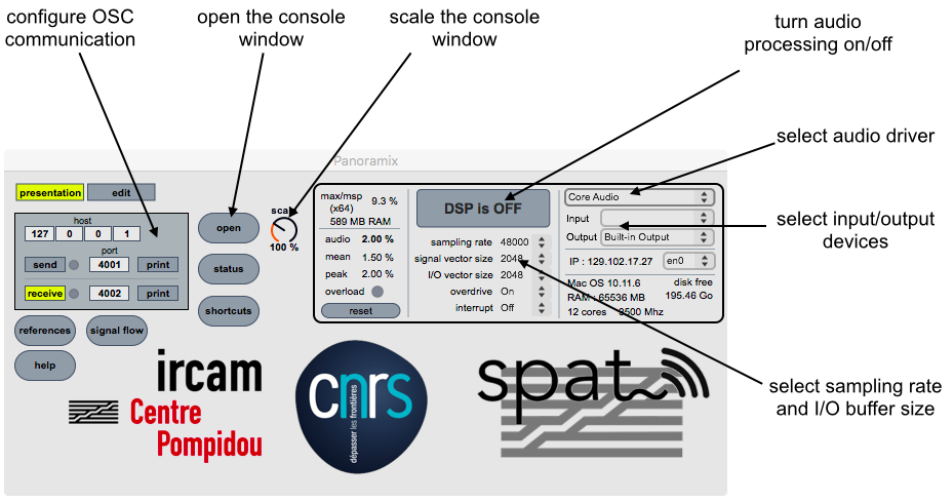

Figure 5: *panoramix* application window

From this point, *panoramix* receives the audio signals from your interface. As previously said, *panoramix* handles up to 128 I/O i.e. it receives the first 128 channels (channels # 1 – 128) from your input device. If your input device supports less than 128 channels, only the first N channels are

actually used. No sound is produced yet, as the console has to be configured now.

- go to the "console window". (Fig. 4)
- create one or several new track(s). (Fig. 6)
- open the input routing window. route the audio input channels (from your audio interface) to the appropriate track(s) and click OK to validate. (Fig. 7)
- inspect the input vu-meters of the tracks; you should now see the audio signal(s) modulating in the appropriate track(s). (Fig. 8).
- create a new bus and choose the spatialization technique you wish (e.g. "Binaural"). (Fig. 9)
- send the track to the bus. (Fig. 10). A spatialization bus is always tied to a reverb bus. At this point, you should see (on the vu-meter) the signal modulating the in spatialization bus and in the reverb bus. However there is still no sound produced. Indeed, only the "Master" track is connected to the output audio driver; one then needs to send the output signals of the bus to the "Master" track.
- route the output of the bus to the "Master". (Fig. 11)
- at this point, you finally see some signals going out of the Master (Fig. 12) and you should hear it (provided that loudspeakers/headphones are connected to the appropriate output channels).

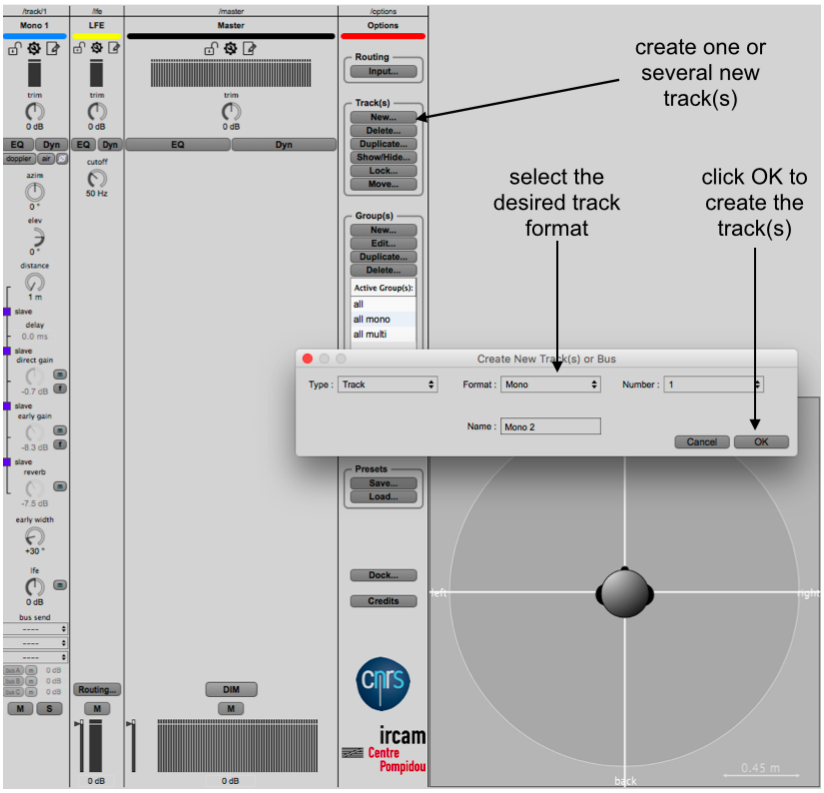

Figure 6: Creating new track(s)

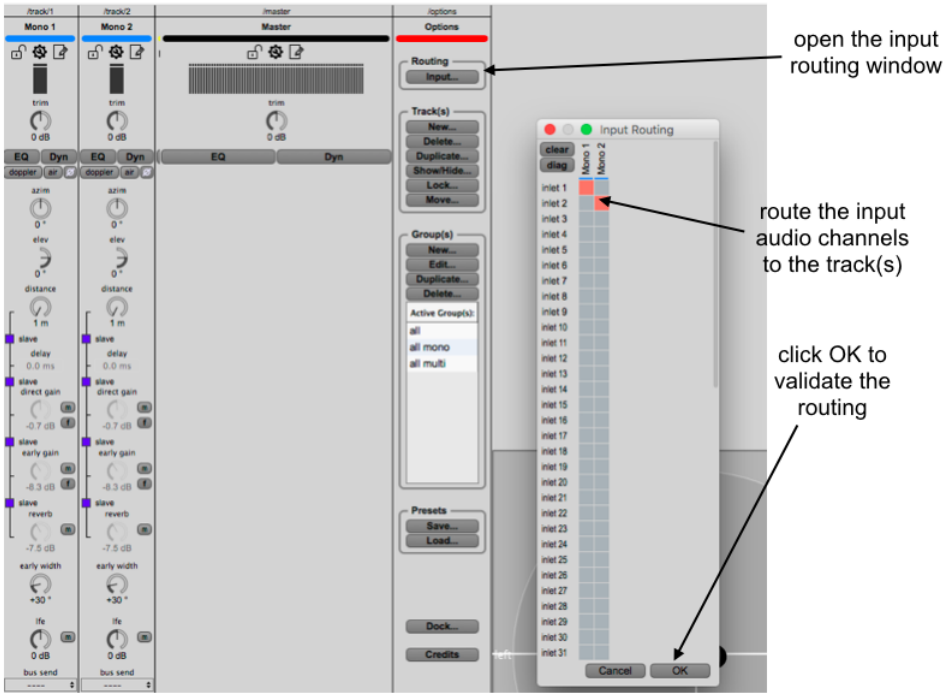

Figure 7: Input routing

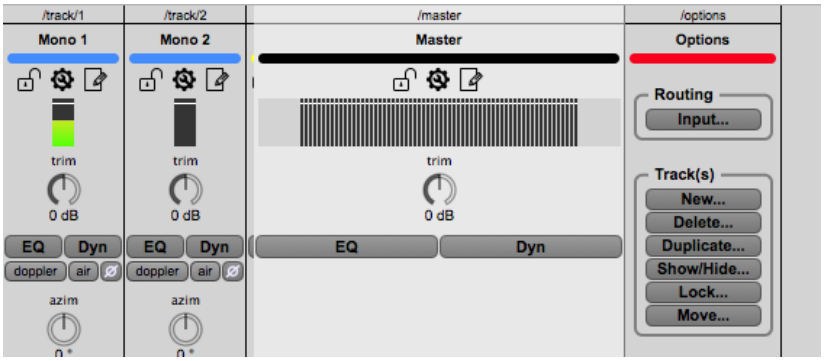

Figure 8: Audio signal received in track #1

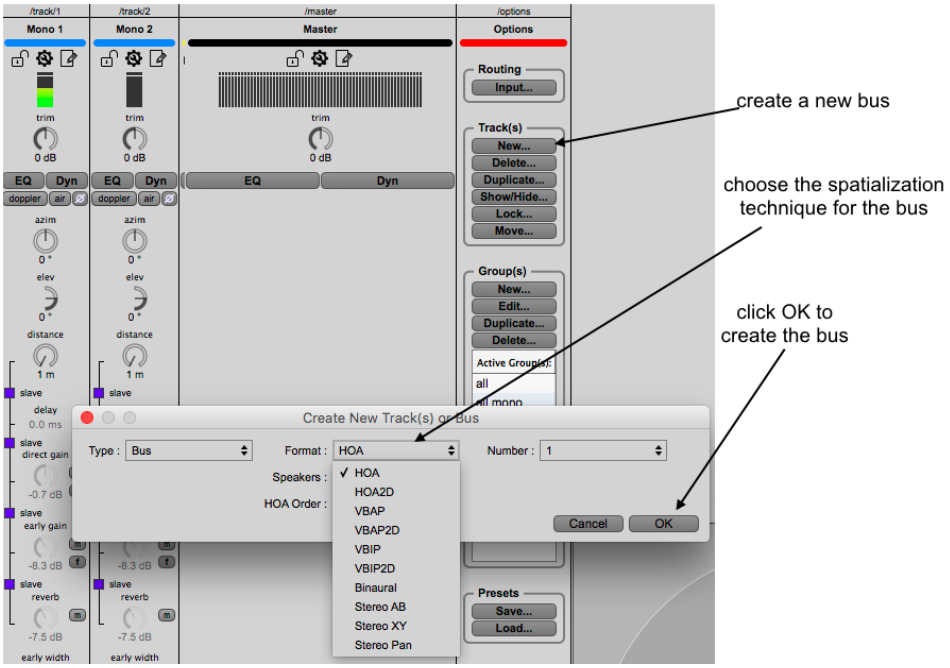

Figure 9: Creating a new bus

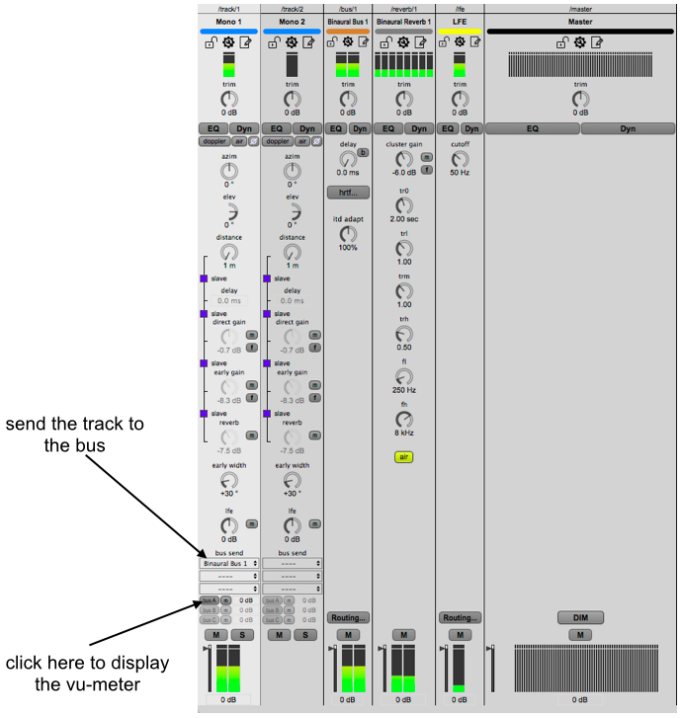

Figure 10: Send the track to the bus

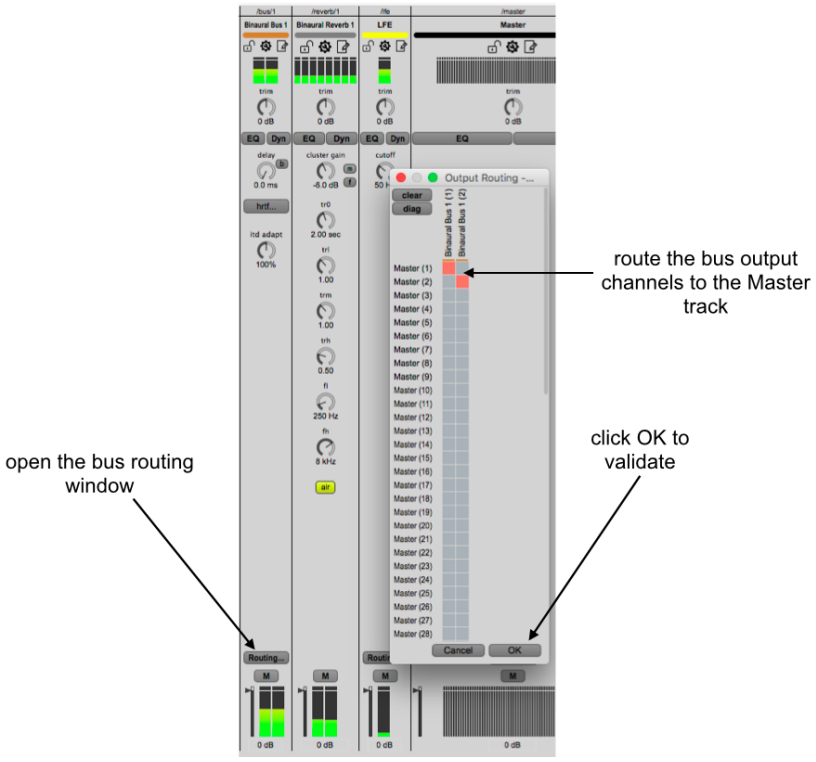

Figure 11: Send the bus to the Master

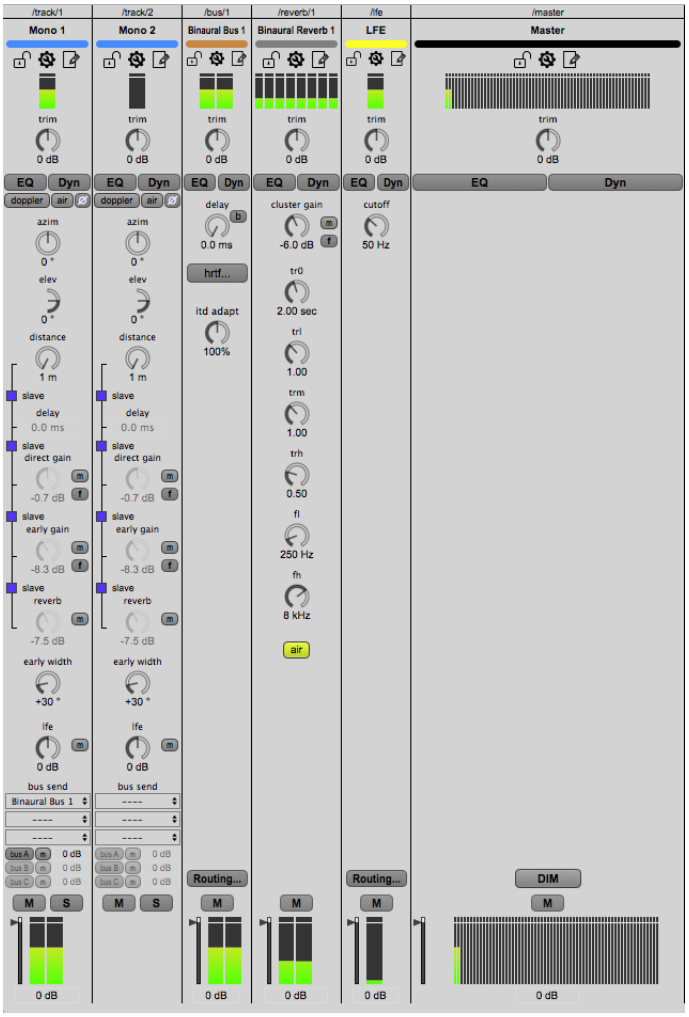

Figure 12: Signals going out of the Master track

4 Working with JackRouter (OSX)

This section provides informations about the "virtual driver workflow" presented in Figure 2 of Section 3. In this example, we consider the use of the JACK Audio Connection Kit for the macOS platform.

- download and install the JACK Audio Connection Kit (Fig. 13). JACK can be downloaded from: http://www.jackaudio.org. This website provides stable versions of JACK. However, if you are using macOS 10.11 (El Capitan), 10.12 (Sierra) or later, you may need to install a beta version of JACK, e.g. JackOSX.0.92_b3 (version used at the time of writing this document). Such beta version can be found online but not necessarily on the official JACK website.
- after installation, it is usually required to reboot the computer.
- open the JackPilot application (usually installed in /Applications/Jack/JackPilot).
- open the JackPilot preferences and configure it. Choose your audio driver, input and output devices, sampling rate, buffer size, and the number of virtual channels (Fig. 14). The number of virtual channels corresponds to the number of audio channels that you'll be able to convey from one application to another (*panoramix*).
- save the preferences and then start the JackPilot.
- open the *panoramix* application. Select "JackRouter" as input and/or output device. Make sure the sampling rate and buffer size match the settings you used in the JackPilot preferences (Fig. 15).
- switch on the DSP in the *panoramix* application.
- open your your digital audio workstation (DAW). In this example, we are using Apple Logic Pro X, but other DAW can be used similarly.
- open the audio preferences of your DAW and select "JackRouter" as input and/or output device. Make sure the sampling rate and buffer size match the settings you used in the JackPilot preferences (Fig. 16).
- now, we have to configure the inter-applications communication, i.e. to manage the virtual connections. This can be done directly using the JackPilot routing interface. Alternatively, you can use "qjackctl" which is a slightly more user-friendly interface for managing the connections.
- open "qjackctl" (usually installed in /Applications/Jack/qjackctl)
- open the connections bay by clicking the "Connect" button (Fig. 17).
- configure the virtual connections as you want. In this example (Fig. 17), we route the outputs of LogicPro to *panoramix* , and the outputs of *panoramix* are sent to the physical outputs ("system").
- create and configure your *panoramix* mixing session as detailed in Section 3.
- do your mix! (Fig. 18).

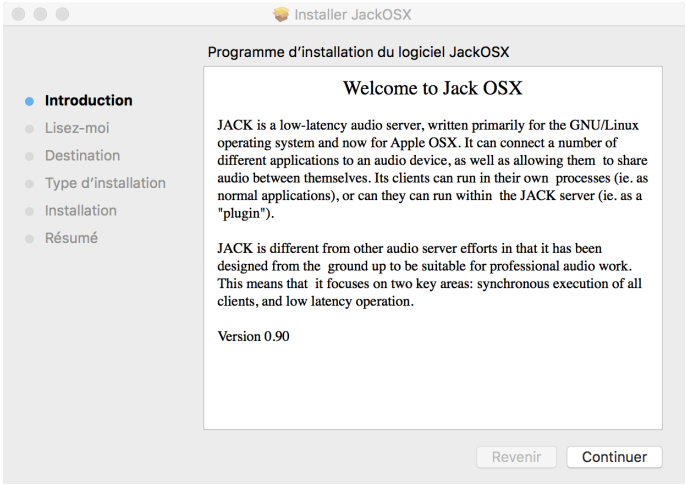

Figure 13: Installing the JACK Audio Connection Kit.

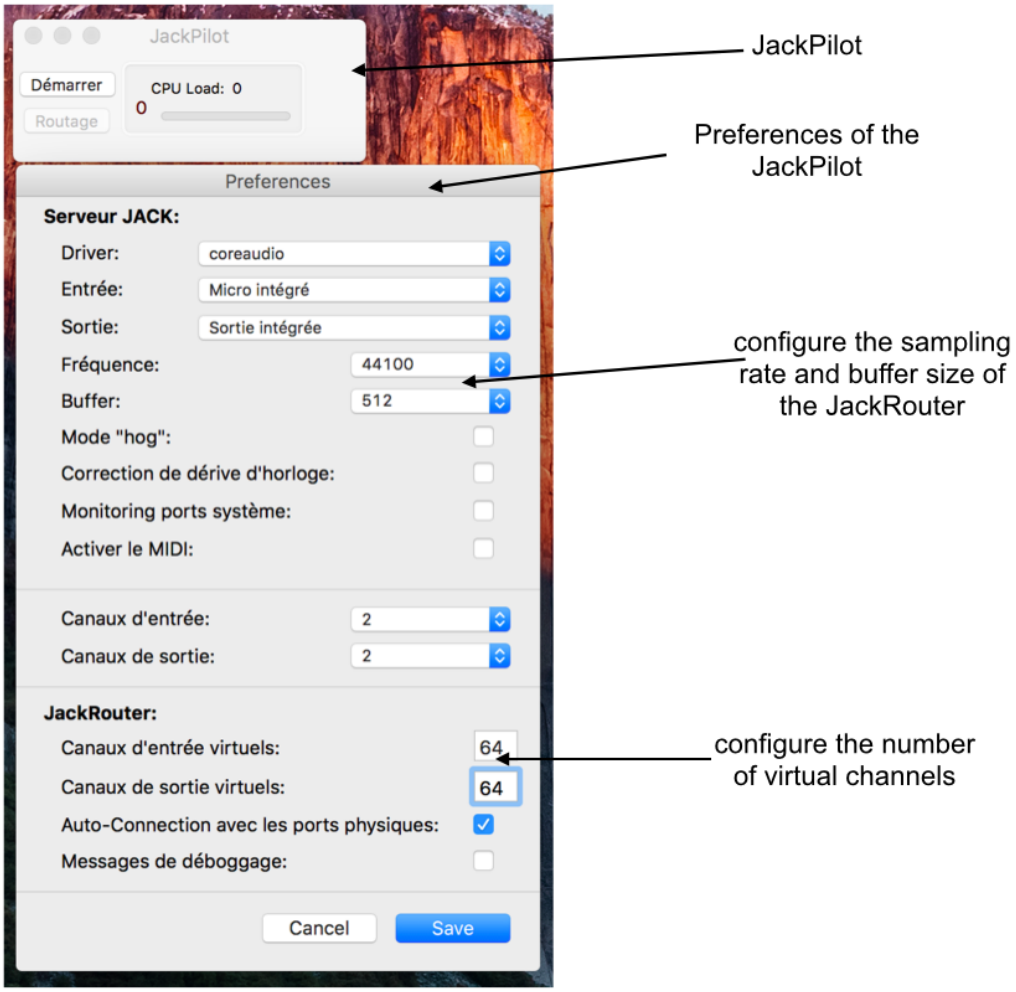

Figure 14: Configure JackPilot.

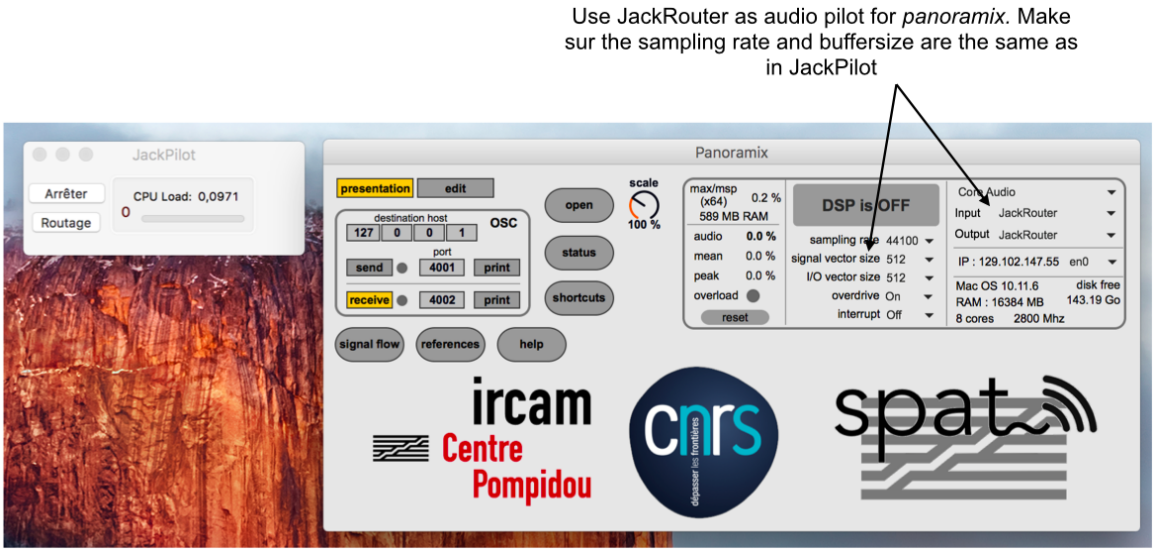

Figure 15: Configure *panoramix* to work with JackPilot.

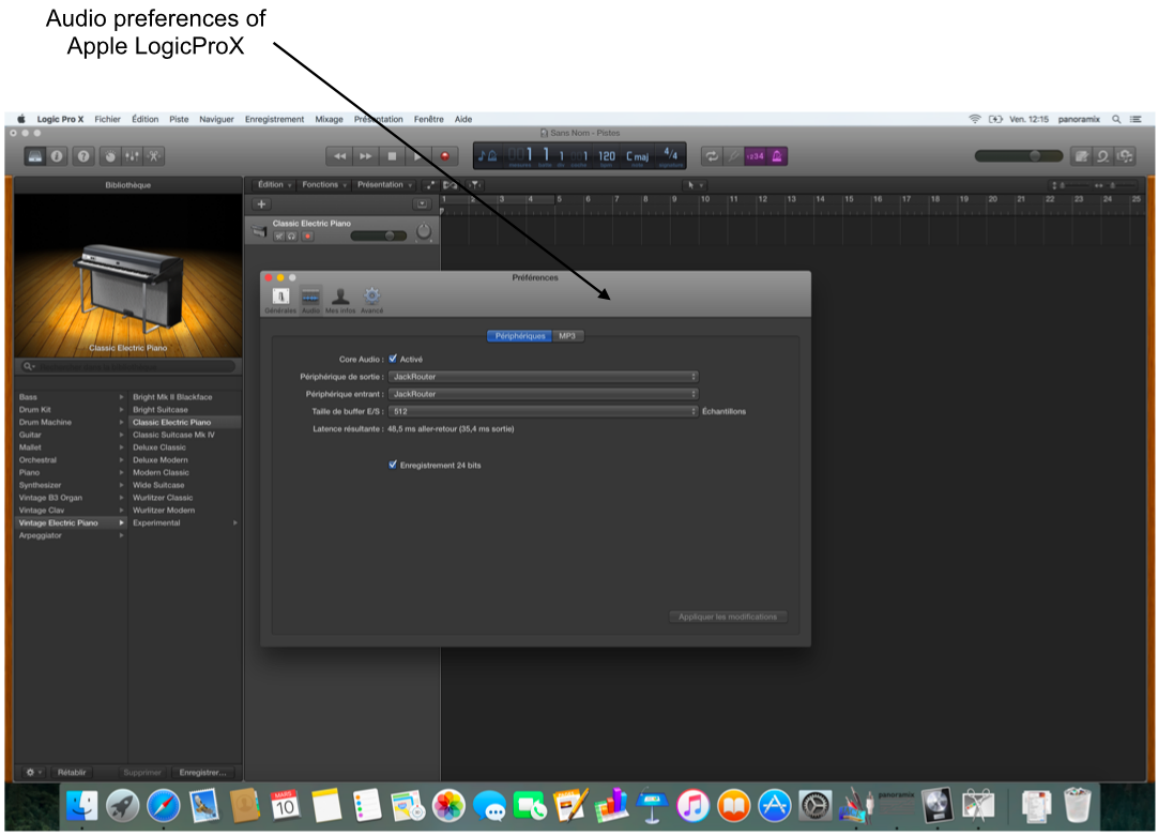

Figure 16: Configure Apple Logic Pro X to work with JackPilot.

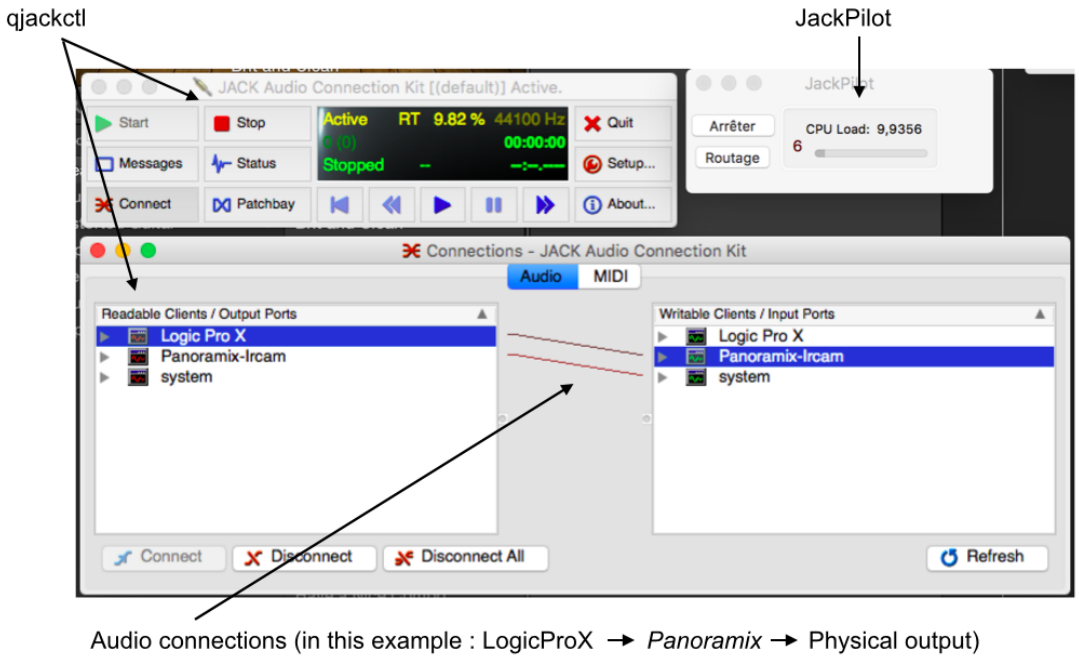

Figure 17: Configure the virtual audio connections.

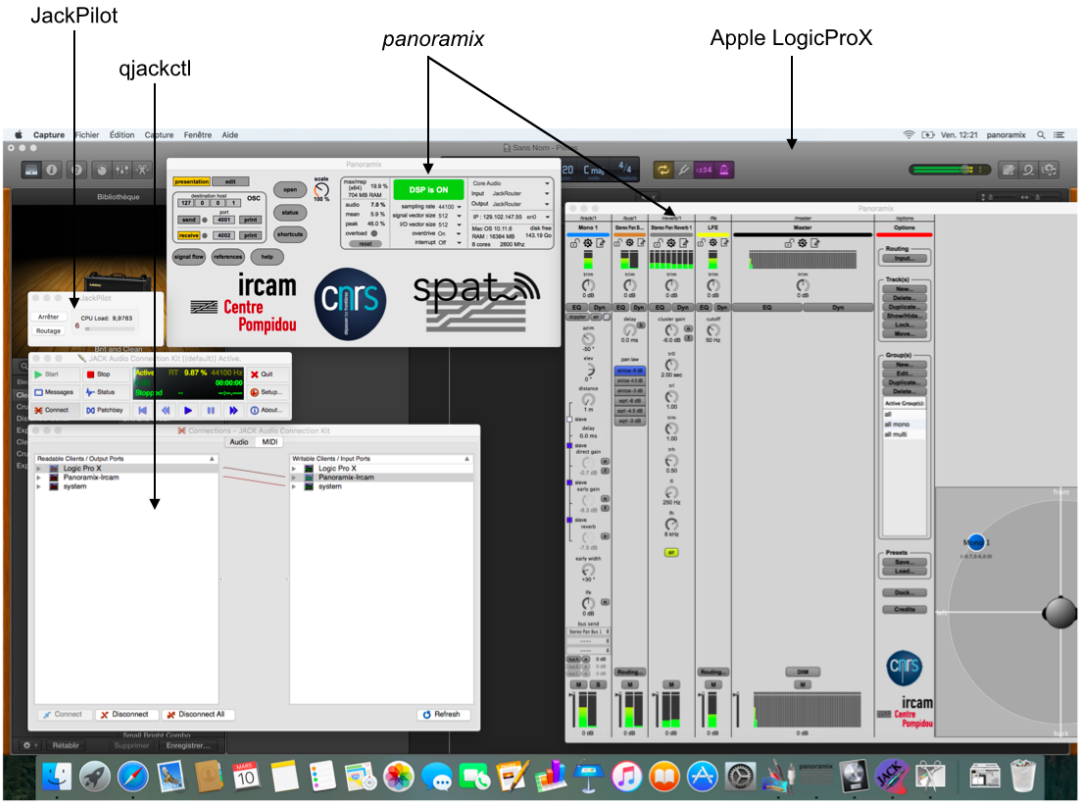

Figure 18: Virtual audio driver workflow: example with *panoramix* and LogicPro virtually connected through the JackPilot.

5 OSC Communication

panoramix can be controlled entirely via the Open Sound Control (OSC) protocol (https://en.wikipedia.org/wiki/Open_Sound_Control).
Open the "status window". (Fig. 19) to discover the OSC syntax for all the controllers. Configure the communication ports in the *panoramix* application window. You can check the "print" toggle to trace the incoming/outgoing OSC messages.

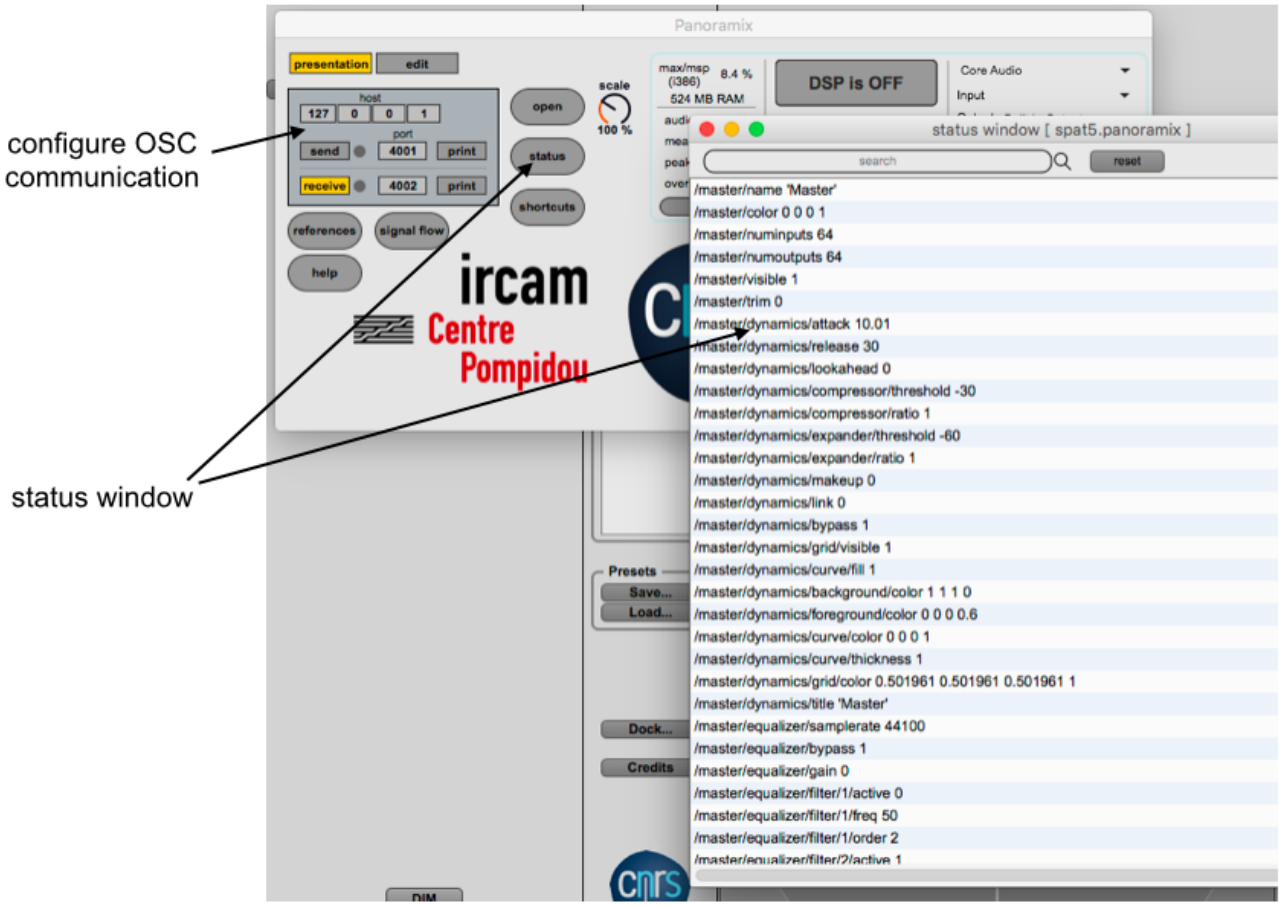

Figure 19: status window

6 HRTF for binaural

For binaural rendering, *panoramix* uses Head-Related Transfer Functions (HRTF) stored in the AES-69 SOFA format [8]. Informations about the SOFA format can be found at https://www.sofaconventions.org. SOFA files are not shipped with the *panoramix* application. A large set of SOFA files is however available in the Ircam Spat package http://www.forumnet.ircam.fr/product/spat-en. *panoramix* currently supports SOFA files under the SimpleFreeFieldHRIR and SimpleFreeFieldSOS conventions.

HRTF data can be loaded in a binaural bus, via the "hrtf..." menu. This button pops up the HRTF selection window (Fig. 20). *panoramix* scans various folders on your hard drive, searching for SOFA files. The scanned folders are:

- ~/Documents/Ircam/sofa
- ~/Downloads
- ~/Desktop
- /Applications/Max 6.1/packages/ircam-spat
- ~/Documents/Max 7/packages/ircam-spat

where "~" is your user home directory. When adding new SOFA files to one of these folders, you may need to click the "refresh" button in the HRTF selection window (Fig. 20). Please note the refresh operation can be lengthy; be patient.
Please note that loading a SOFA file in a binaural bus will apply the same HRTF file to all tracks attached to this bus.

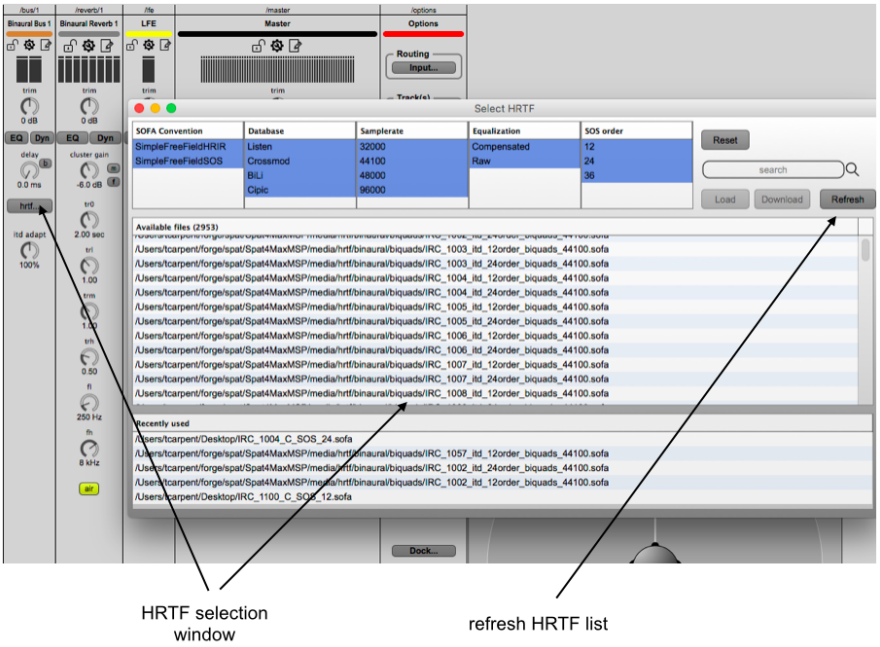

Figure 20: HRTF selection window

Credits

Design and Development: Acoustics and Cognition Research Group / Ircam.

With the great help of: Clément Cornuau, Mélina Avenati, Julien Aléonard, Luca Bagnoli, Sylvain Cadars, Jérémie Henrot.

panoramix builds on Ircam Spatialisateur (**Spat**∼) modules. The design of **Spat**∼ and the reverberation module are protected under different French and international patents ([FR] 92 02528; [US] 5,491,754, [FR] 95 10111; [US] 5,812,674).

References

[1] Thibaut Carpentier. ToscA : Un plugin de communication OSC pour le mixage spatialisé orienté objet. In *Journées d'Informatique Musicale*, Montreal, Canada, May 2015. https://hal.archives-ouvertes.fr/hal-01247477v1.

[2] Thibaut Carpentier. ToscA: An OSC Communication Plugin for Object-Oriented Spatialization Authoring. In *Proc. of the 41st International Computer Music Conference*, pages 368 – 371, Denton, TX, USA, Sept. 2015. https://hal.archives-ouvertes.fr/hal-01247588v1.

[3] Thibaut Carpentier. Panoramix: 3D mixing and post-production workstation. In *Proc. 42nd International Computer Music Conference (ICMC)*, pages 122 – 127, Utrecht, Netherlands, Sept 2016. https://hal.archives-ouvertes.fr/hal-01366547v1.

[4] Thibaut Carpentier and Clément Cornuau. panoramix: station de mixage et post-production 3D. In *Journées d'Informatique Musicale*, pages 162 – 169, Albi, France, April 2016. https://hal.archives-ouvertes.fr/hal-01300314v1.

[5] Thibaut Carpentier, Markus Noisternig, and Olivier Warusfel. Twenty Years of Ircam Spat: Looking Back, Looking Forward. In *Proc. of the 41st International Computer Music Conference*, pages 270 – 277, Denton, TX, USA, Sept. 2015. https://hal.archives-ouvertes.fr/hal-01247594v1.

[6] Jean-Marc Jot. Real-time spatial processing of sounds for music, multimedia and interactive human-computer interfaces. *ACM Multimedia Systems Journal (Special issue on Audio and Multimedia)*, 7(1):55 – 69, 1999.

[7] Jean-Marc Jot and Olivier Warusfel. A real-time spatial sound processor for music and virtual reality applications. In *International Computer Music Conference*, pages 294 – 295, Banff, 1995.

[8] Piotr Majdak, Yukio Iwaya, Thibaut Carpentier, Rozenn Nicol, Matthieu Parmentier, Agnieszka Roginska, Yôiti Suzuki, Kanji Watanabe, Hagen Wierstorf, Harald Ziegelwanger, and Markus Noisternig. Spatially Oriented Format for Acoustics: A Data Exchange Format Representing

Head-Related Transfer Functions. In *Proc. of the 134th Convention of the Audio Engineering Society*, Roma, Italy, May 4-7 2013.