Spatial audio hardware & Software solution for Tilburg University's VR Cave.

INSTRUCTIONS FOR WINDOWS 10

Package Contents:

RAR download includes:

- Spatial Engine VR Ableton m4l Plugin
- Jack Audio Connection Kit 1.9.11
- Dependencies folder: Spat5-x64
- Ableton Live example project
- Ambisonic Example files (AmbiX)
- Unity3D package
- Documentation
  Running Spatial Engine inside Ableton Live

Running Spatial Engine inside Ableton Live

And setting up the 42.2 Genelec spatial audio system with Unity3D

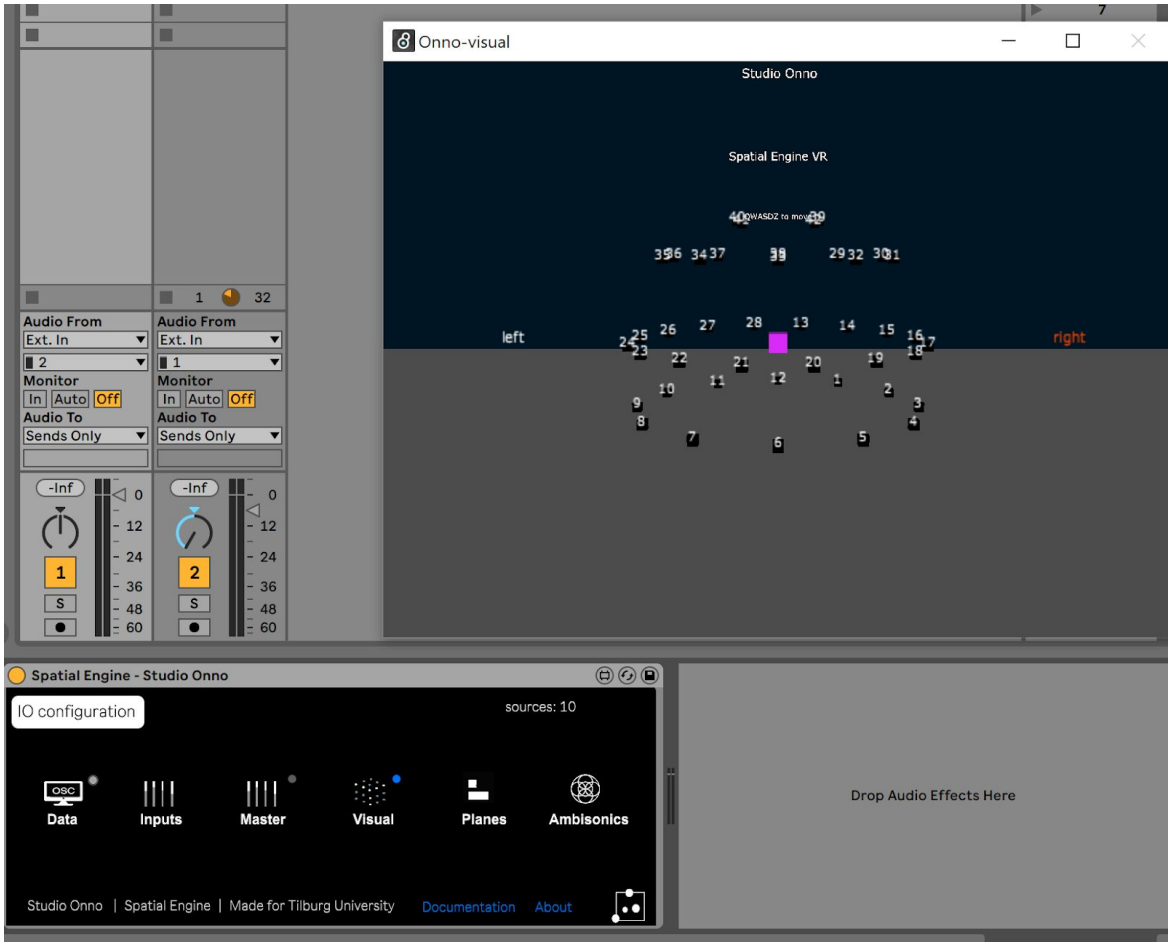


Table of content

0 Introduction

In these instructions we will go over the basics for installing, and Using Spatial Engine as a Max for Live plugin. The goal of this system is to create a synchronized way to spatialize audio sources on the 42.2 Genelec speaker system. Audiosources can be placed virtually anywhere on the Speaker system. Audio will be streaming live from Unity3D's audio mixer Asset. The position of soundsources is directly translated into speaker volumes, phase differences, and adaptive reflections by our software: Spatial Engine VR. This software is a streamlined implementation of Ircam's Spat5 Library; a vast toolkit for spatial audio algorithms. We optimized Spatial Engine VR to be used with this 42.2 Genelec system, and in the context of Virtual Reality. To improve the stability of our spatial audio solution, we opted to outsource the audio driver handling to Ableton Live. This Ableton Live integration brings not only stability, but also a great many new options to the table such as individual source processing and IO management.

1 Preparations

Installing Dependencies

1.1 Install Ableton Live Suite
> Ableton Live for students and teachers | Ableton

- Make sure to get the full version: Live Suite. including Max for Live.

1.3 Asio driver for Digigram Hardware
> Production & Sound Cards - Digigram

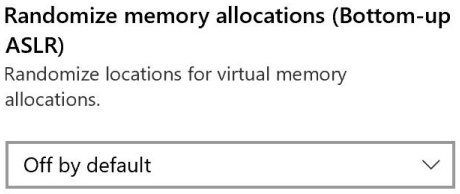
- On the Digigram website: Find the correct ASIO Driver for your specific soundcard.

1.4 Microsoft Visual C++ Dist. Kit.
> The latest supported Visual C++ downloads (microsoft.com)

- It also ships with Visual Studio, so chances are you already own a version.

Making necessary changes to your Windows System

1.5 Disable randomize memory allocations (Bottom Up ASLR)
> Open Windows Security
> Open App and Browser Control
> Scroll down to Settings for exploit protection
> Disable: Random memory Allocations (Bottom Up ASLR) > Restart your Machine to implement the changes.



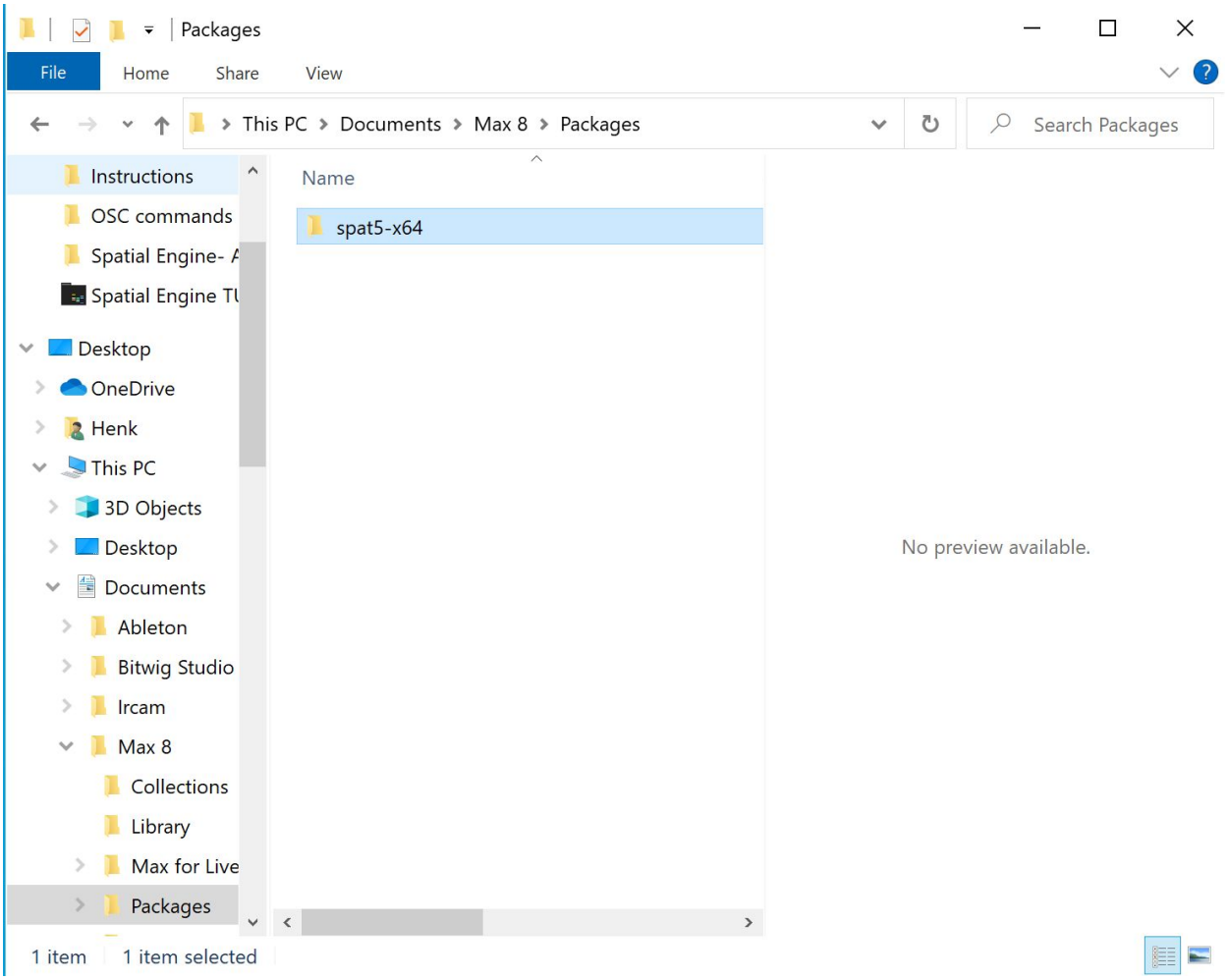
Bottom Up ASLR encrypts inter-app communication. For this reason, it is problematic while setting up an audio connection from Unity3D to Ableton Live. For this reason, it must be disabled.

1.6 Implement changes
> Reboot your PC now

2 Spat5-x64 MXO library

This Onno_Spatial_Engine package comes with Spat5-x64. Ableton needs to find this library in order to run Spatial Engine successfully.

2.1 Allocate the spat5-x64 folder
> Navigate to : Spatial Engine- Ableton Plugin 18-02-2021\Dependencies
> Copy “Spat5-x64” folder to: C:\Users\User\Documents\Max 8\Packages



Ableton (Max for Live) will automatically scan this folder for packages.

I don’t see this folder?
> Make sure Ableton Live suite is installed
> Reboot your system if needed

3 Install and Configure J.A.C.K

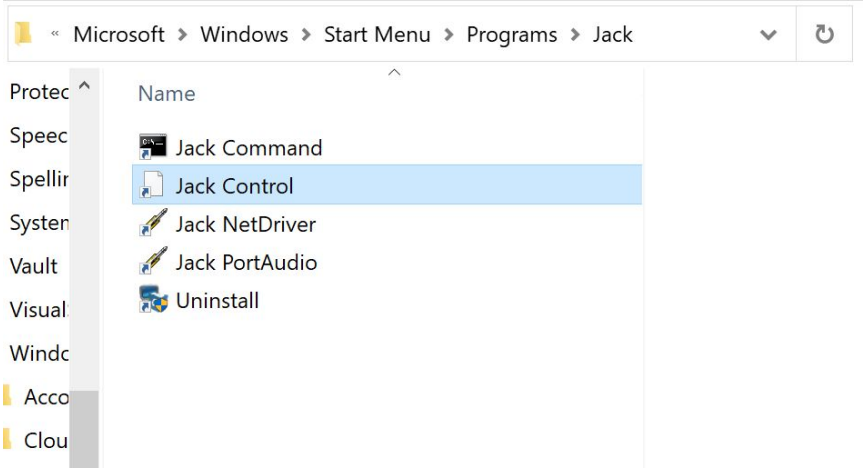
Jack Audio connection kit is an open-source realtime audio server for streaming audio between applications. We will use Jack to send multichannel audio from Unity3D to Ableton Live. To set up Jack correctly for use with unity3D, follow these steps carefully.

3.1 Download and Install Jack_v1.9.11_64_setup.exe
https://github.com/jackaudio/jackaudio.github.com/releases/tag/1.9.11
(for other versions of Jack, Unity3D Jack Send Plugin must be recompiled from source)
Original Windows instructions here: https://jackaudio.org/faq/jack_on_windows.html Our instructions differ slightly to make Jack work in sync with Unity3D.

3.2 DLL Register
JackRouter.dll file must be added to the registry.

> Unregister previous versions before installing a new one!
> Open an elevated command prompt
> Enter the registration: regsvr32 "C:\Program Files (x86)\Jack\64bits\JackRouter.dll"
- Should respond with: "Successfully registered JackRouter.dll"

Navigate to Windows > Start Menu > Programs > Jack



We will use two of these applications:
Jack PortAudio
- For physically connecting to the soundcard

Jack Control
- To control the server, and make audio connections

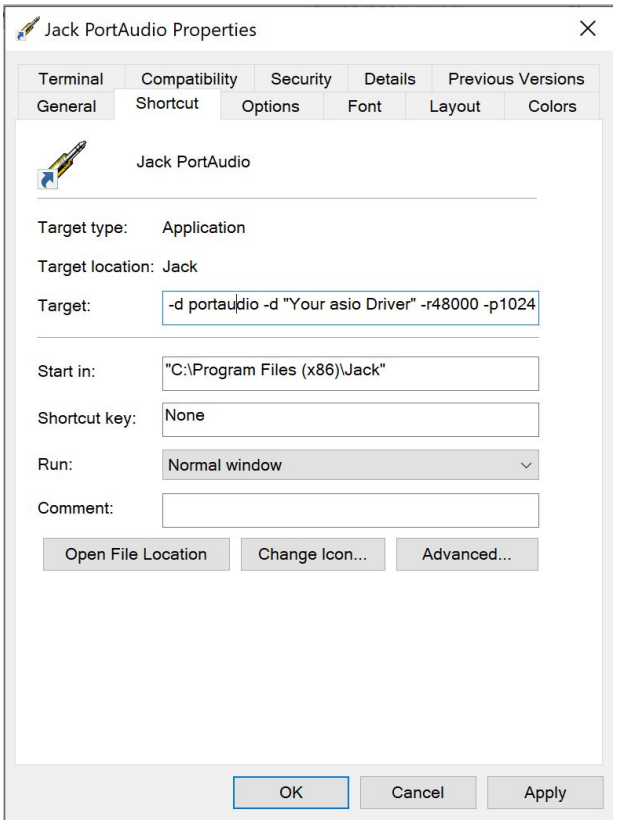
3.3 Finding the Digigram Asio driver
> Open an elevated command prompt
> Navigate to the Jack installation folder:
`cd C:\Program Files (x86)\Jack`
> Run this command:
`jackd -d portaudio -l`

This will present a list of all available audio devices/drivers on your system.

> Find Digigram ASIO LXIP in the list
> Copy the exact name of the driver, we will need it later.

3.4 Setting Backend params
> Com back to: Windows > Start Menu > Programs > Jack
> Select Jack PortAudio, and make a shortcut on the desktop.
> Right mouse click on the shortcut and select “properties”
> Navigate to shortcut > Target

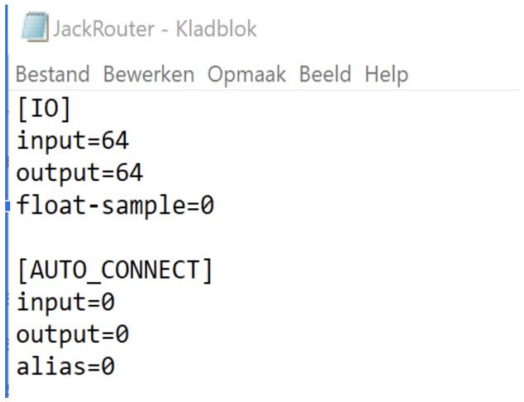
> Paste the following line in the target area:
"C:\Program Files (x86)\Jack\jackd.exe" -R -S -d portaudio -d "Your Digigram ASIO" -r48000 -p1024
> replace “your Digigram Asio driver” with Digigram's ASIO Driver. (within the quotations)



Information about Jack PortAudio - Backend params
PortAudio is a Utility to launch the jackserverwith backend params. It is possible to pre-define an asio device, set SampleRate, and Buffer Size, and make other prefixes. For a complete overview of all backend params: *jackd: JACK Audio Connection Kit sound server - Linux Man Pages (1) (systutorials.com)*

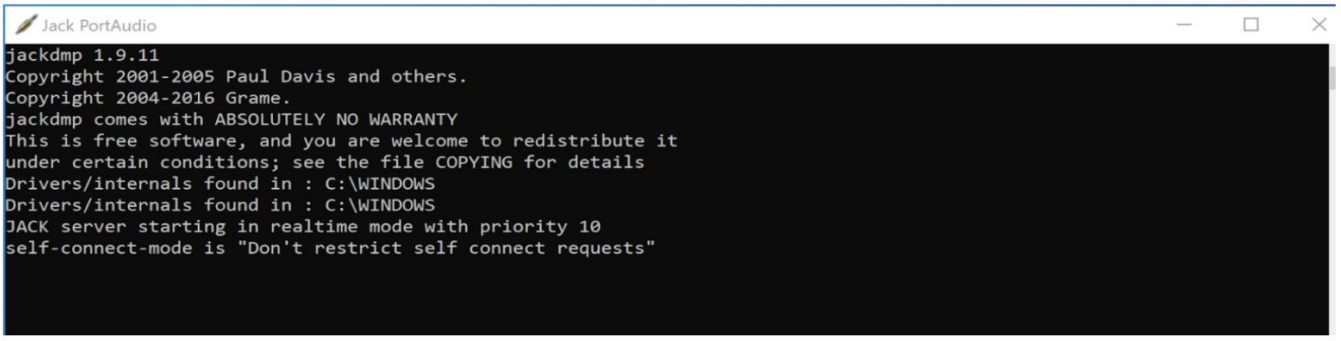
> Click “apply”
> Click “OK”

3.5 Configure the JackRouter.INI File for 64 audio channels
> Open C:\Program Files (x86)\Jack\64bits\JackRouter.INI as admin
> Set the IO to 64 channels, and turn off auto connections:



> Save the file

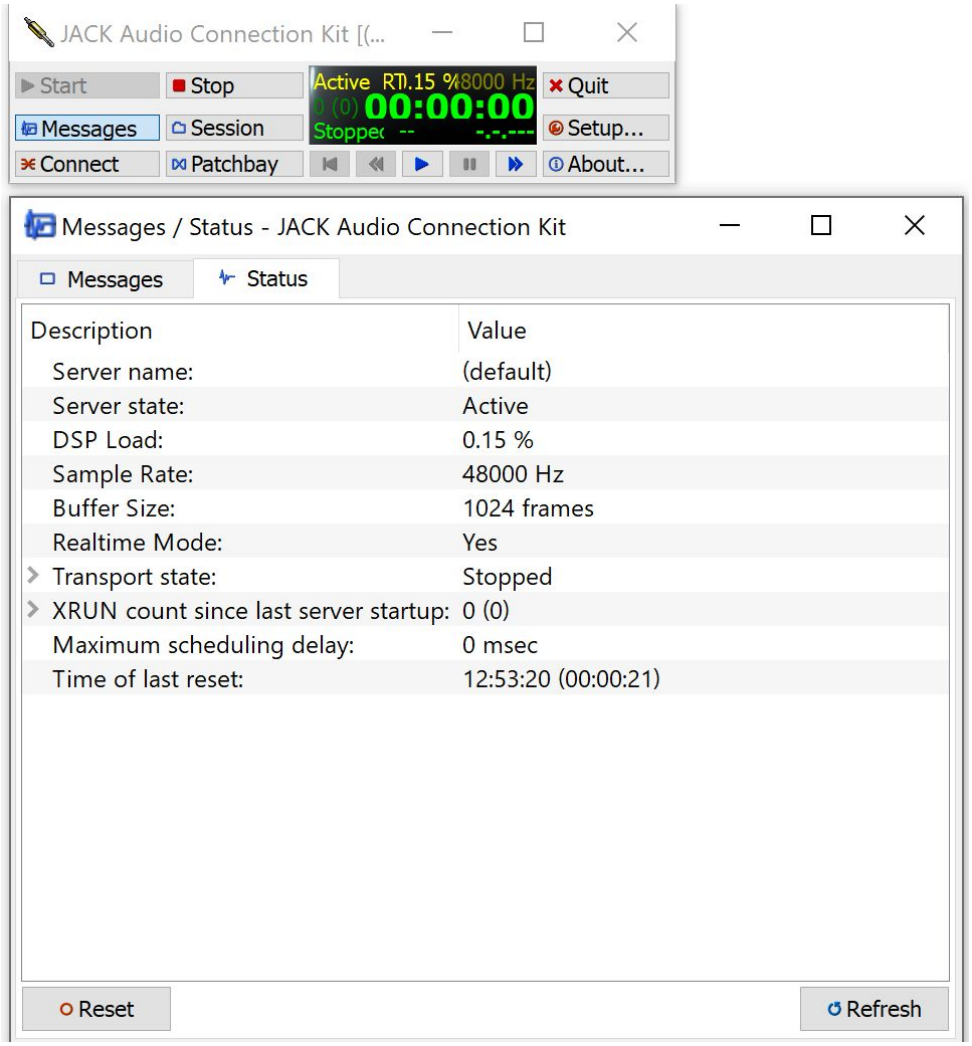
3.6 Launching JackD server
> Launch Jack PortAudio (**first**)
- It should look like this:



Jack server is now running with the right settings for linking to Unity3D

3.7 Make a shortcut for Jack Control.exe on the desktop
> Navigate back to: Windows > Start Menu > Programs > Jack
> Select Jack Control
> Create a shortcut on the desktop
> Launch Jack Control (**second**)
- The server is now running with the right settings for linking to Unity3D .

3.8 To verify that the server is running correctly:
> Click “messages” in the JackControl menu
> Navigate to the “Status” tab



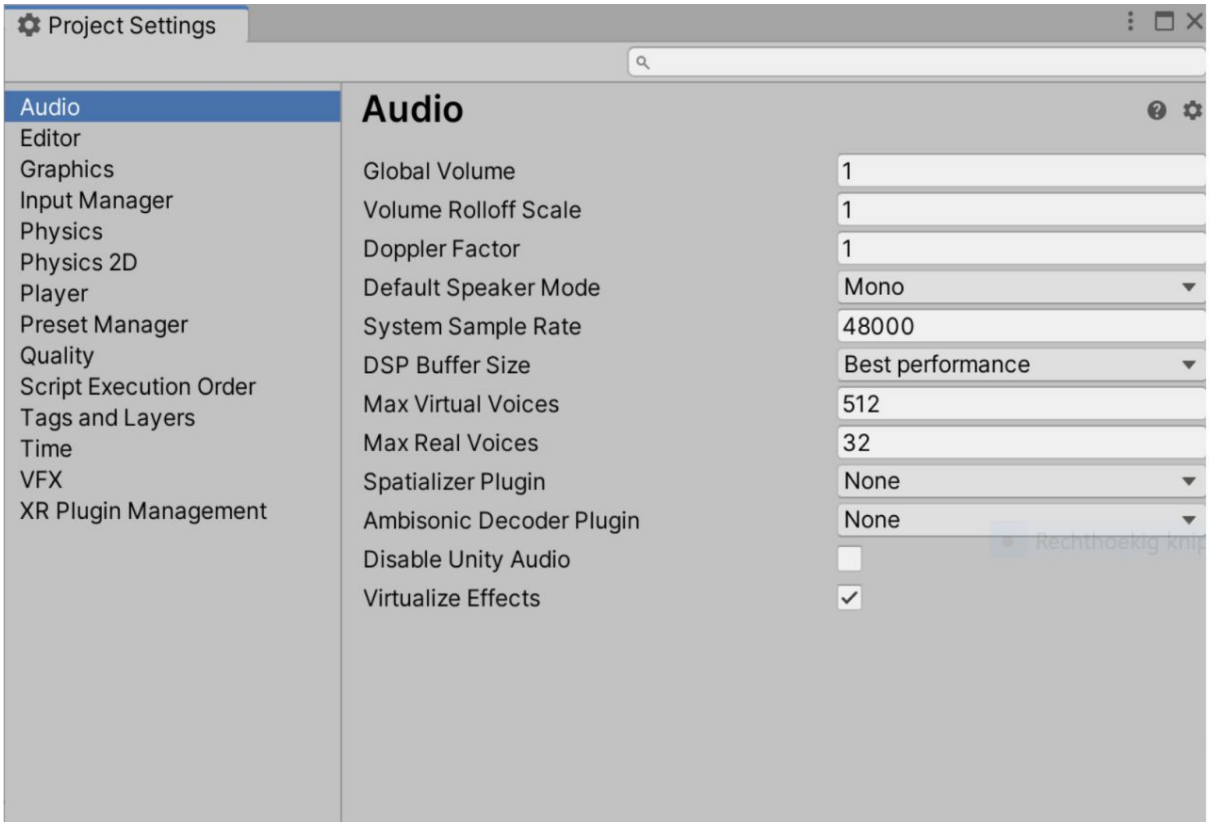
Make sure all these values are the same:

- State: Active
- DSP load (should not be to high)
- Sample rate: 48000hz
- Buffersize: 1024 frames
- Xrun count should always be 0

Jack Server is now running correctly!
> Leave Jack running for the remaining steps.

4 Setting up Unity3D for Linking to Jack.

4.1 Unity3D project audio settings
> Start a new Unity3D project.
> Navigate to: Edit > Project settings > Audio.
> Match these settings:

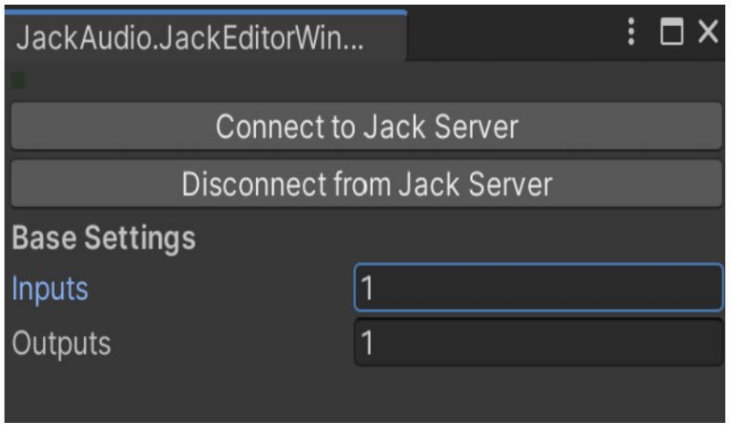


The default **block size** for Unity3D is 1024 samples This is equal to "DSP buffer size: best performance mode".
- Default **Speaker Mode** is set to **mono** because Jack sources are mono.
- The default System **Sample Rate** for Unity3D is 48000 hz. Our JackServer is running at this rate. it is important these numbers match.
- We don't need a spatializer plugin because we are doing the spatialization outside of Unity3D.

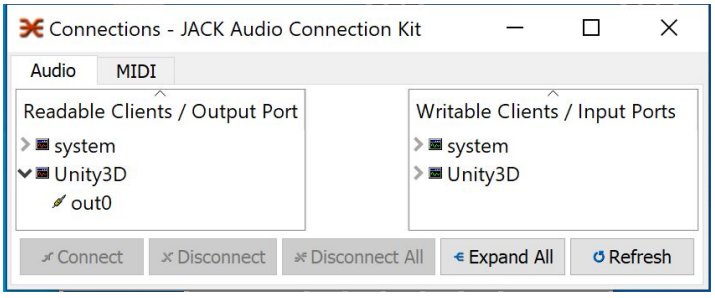
4.2 Import Unity Package
> In Unity: Navigate to Assets > Import package > Custom Package.
> Import Onno_spatial.unitypackage

Package contents
- Jack Audio for Unity mixer Plugin.
- OSC Scripts: For communication with Spatial Engine VR.
- “Hello World!” Example Scene

4.3 Connect Unity3D to Jack
> In Unity3D: Navigate to: Window > Jack.
> Set the number of inputs and outputs and click: "Connect to Jack Server"
- For the "Hello World" example, we are just using 1 source.

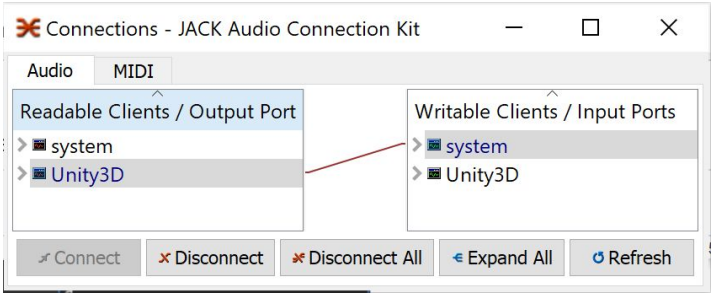


> Unity3D now appears in: Jack Control > Connect.



When Unfolding the "Unity3D" tab, one output is visible. This corresponds with the 1 output we set in the Unity3D Jack Editor Window (Jack Multiplexer Script).

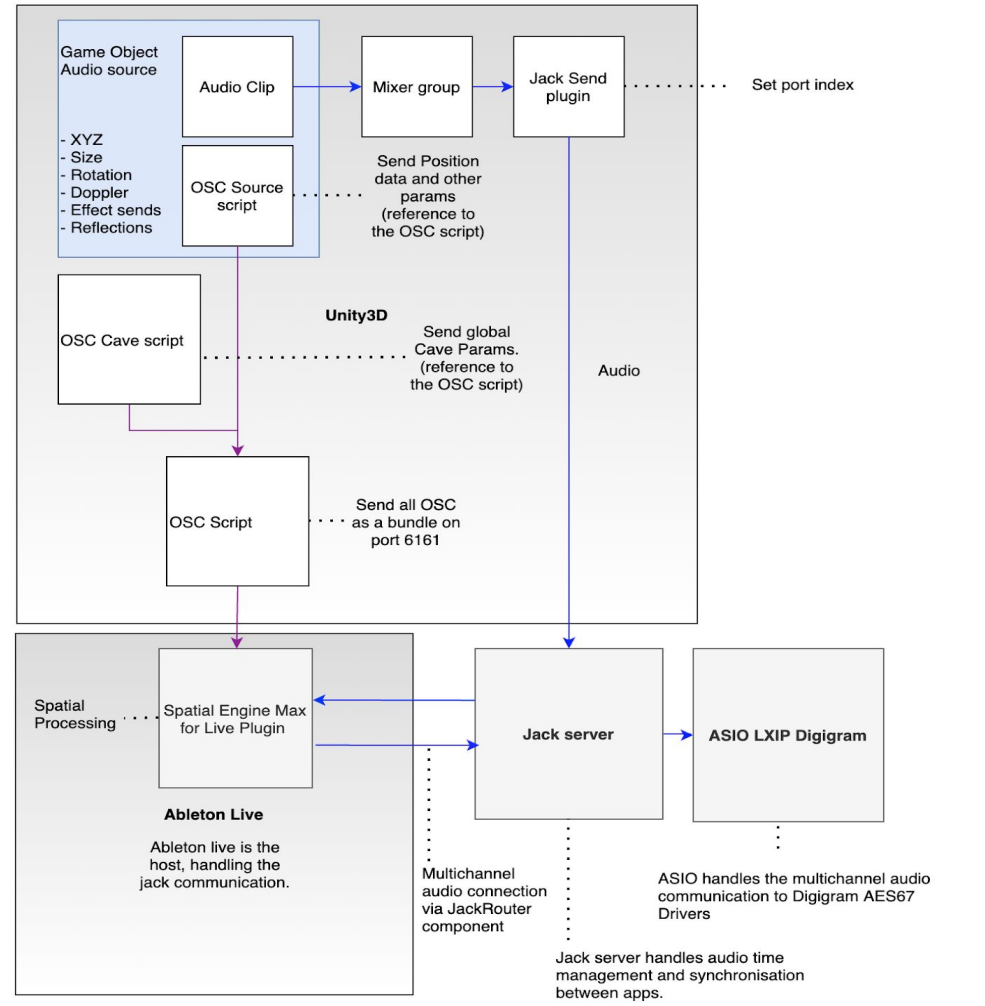
4.4 Connect Unity3D to System in Jack Control > Connect
> In Jack: Connect Unity3D Output port to "System Input port" Like this:



- The System **output ports** transmit audio to the physical card using ASIO
- A **Client** Is an application that can speak to jack via Native Bindings, or JackRouter.
- A **Port** is a single mono channel in the multichannel audio stream

Everything is now set up to test out the example scene!

5 Hello World Example scene



- **AudioSource GameObject "Source 1”**
- Audio clip, sending audio to Mixer group 1
- OSC Source script: Spatial audio source params

- **Cave Audio Master**
- OSC script: Send OSC data to Spatial Engine VR.
- OSC Cave script: Control Spatial Engine VR main cave parameters

- **Audiomixer Asset “Jack"**
- Master Bus
- Mixer group “Source 1” receiving audio from "source 1”
- Jack Send audio mixer Plugin: Sends audio to the first input channel in Jack audio.

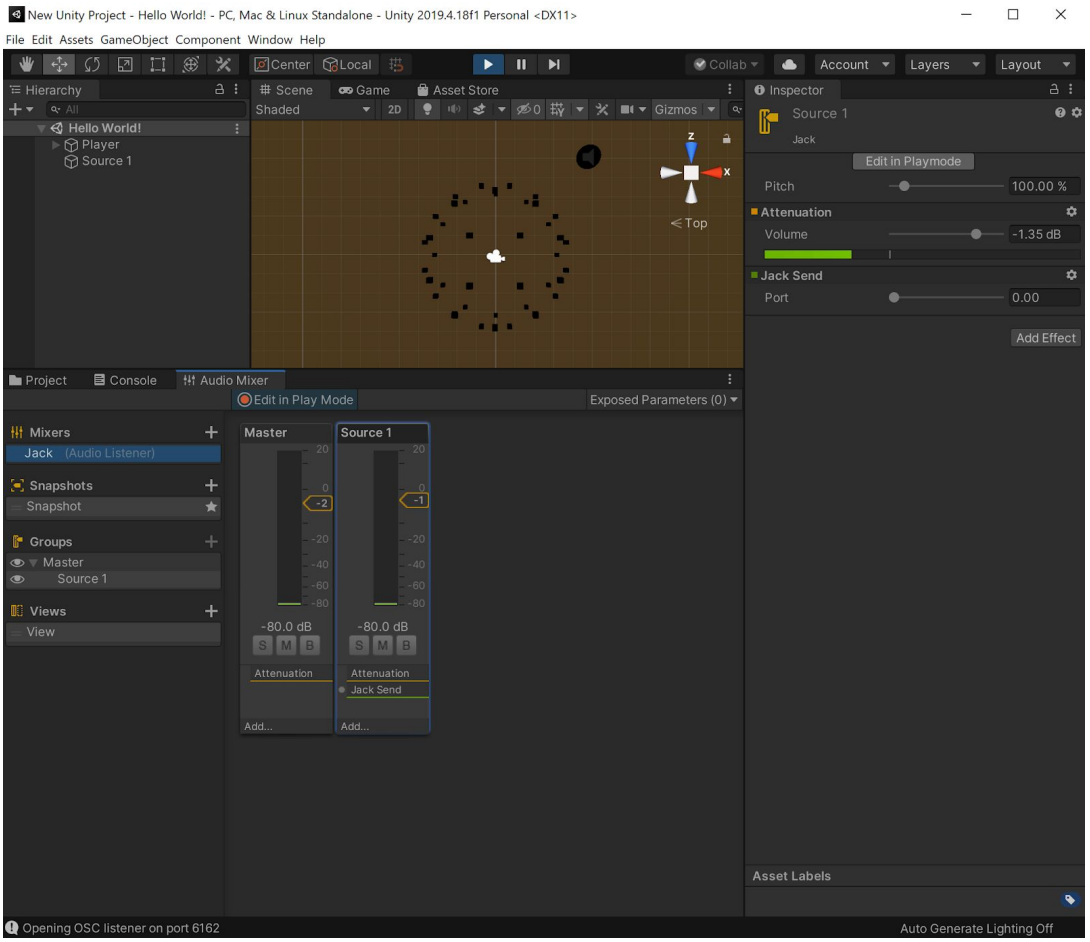
- **Spatial Listener**
- All source positions are relative to the spatial listener
- Display the loudspeakers as a static grid around the player.

5.1 Load the example scene

> In Unity3D: Navigate to: Assets > Onno_Spatial > Scenes

> Open the HelloWorld.unity example scene. "HelloWorld" Example scene

5.2 Audio signal test



> open: Unity Audio mixer Asset > Mixers > Jack

> Navigate to the inspector window for “mixer group source 1”

> Make sure the “jack send” plugin is configured to “port 0”

- **Note: Multiple audio signals to the same port will result in audio problems**
- Every jack send plugin insertion in Unity3D needs a new port (0, 1, 2, 3, etc.
- It is important that the order of ports is chronological)

> Run the Unity3D simulation!

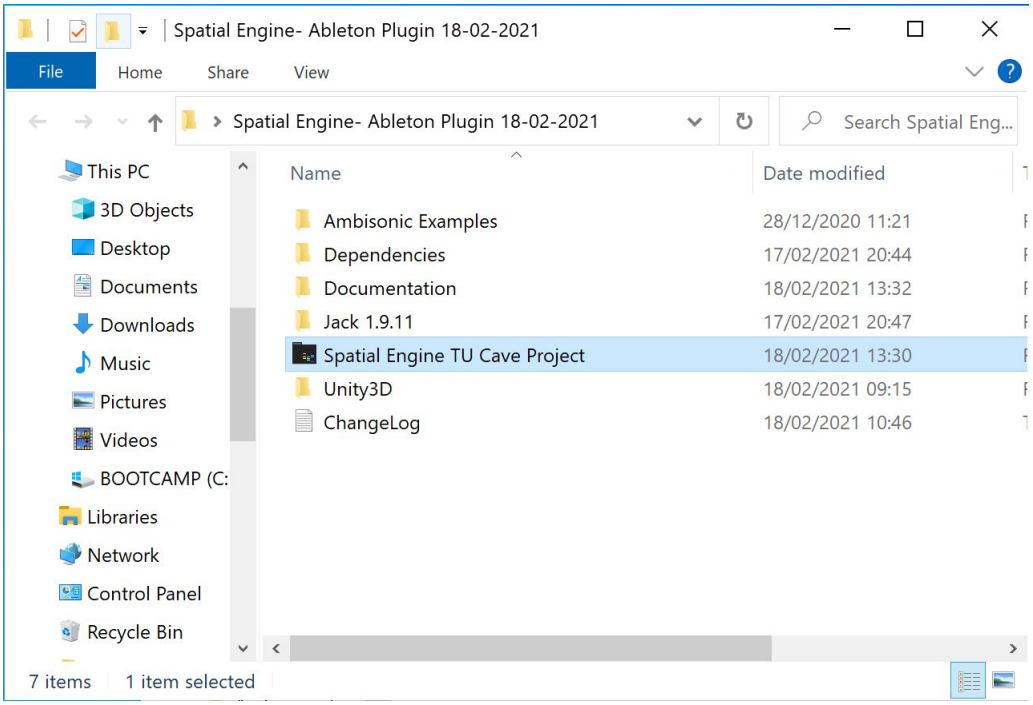
If all steps are done correctly, audio should be flowing out of Genelec speaker 1

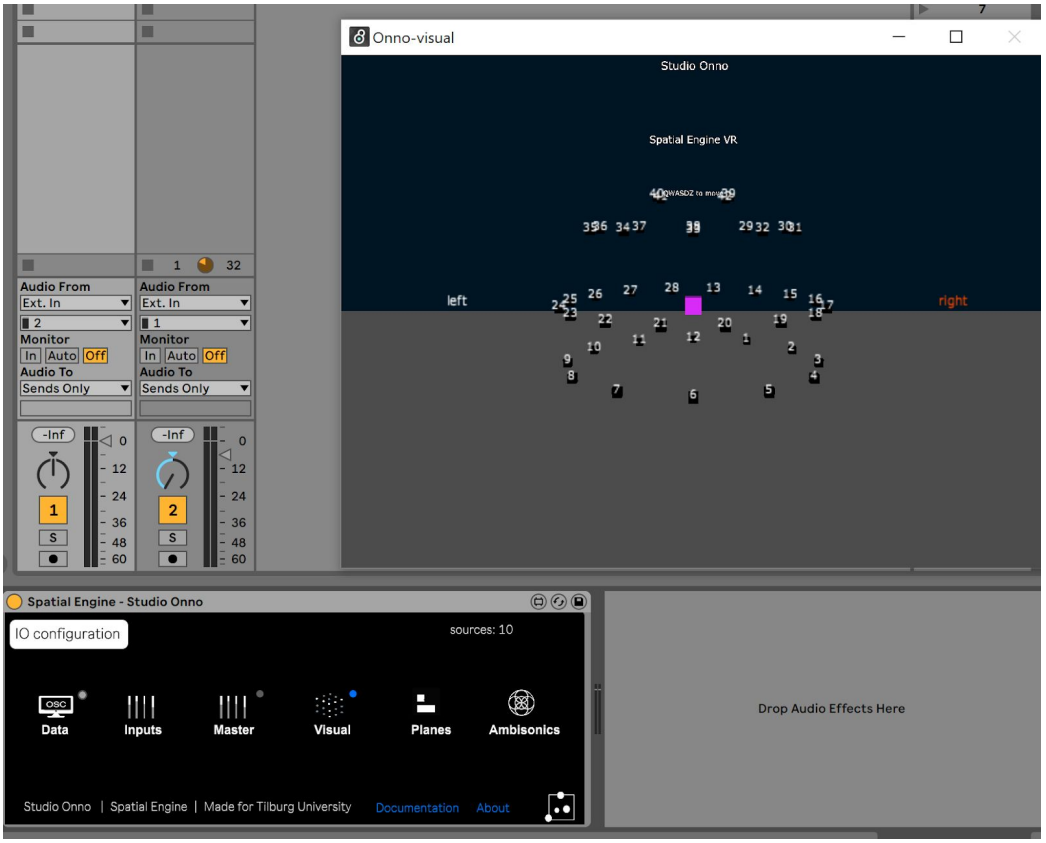
6 Connecting Ableton Live to Jack

6.1 Ableton Example project

> Open the Spatial Engine TU Cave Project folder

> Open Spatial Engine TU Cave Project.als

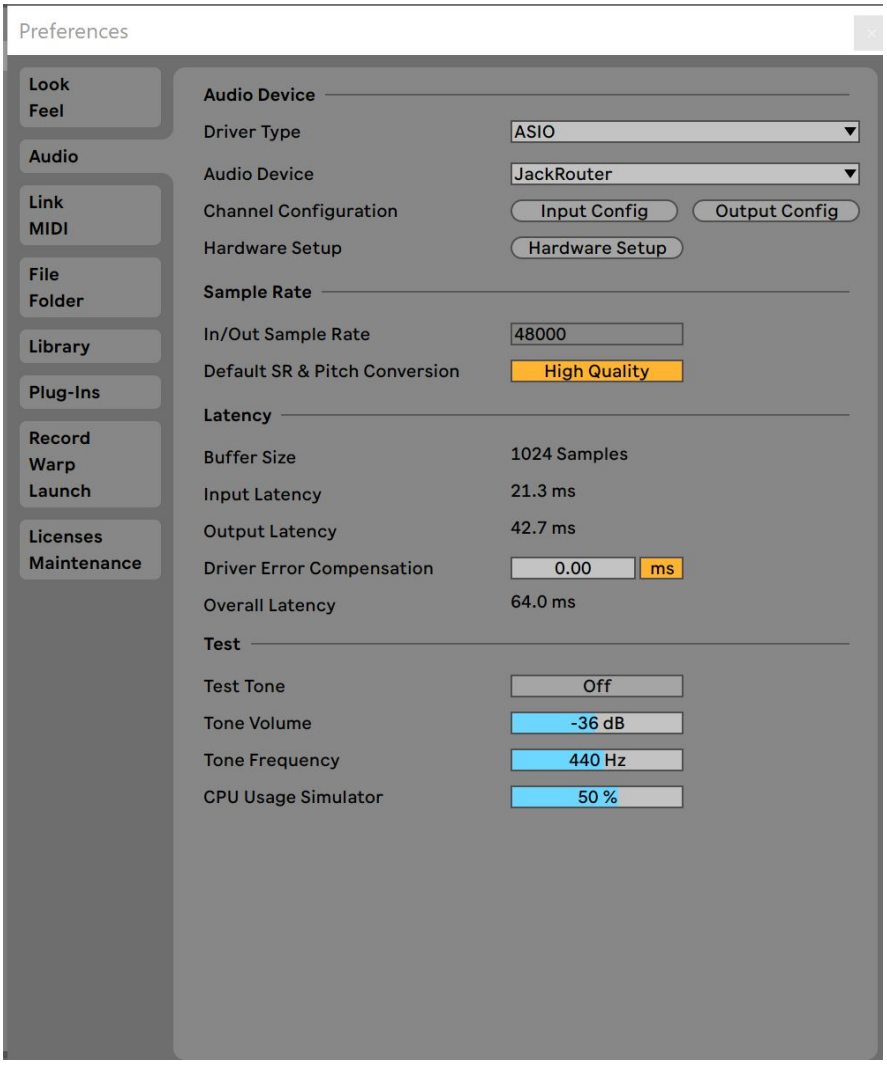




6.2 Ableton audio driver preferences
> In ableton: Navigate to Options > Preferences > Audio
> Set “Driver type” to “ASIO”
> Set “Audio Device” to “JackRouter”

I don’t see JackRouter?

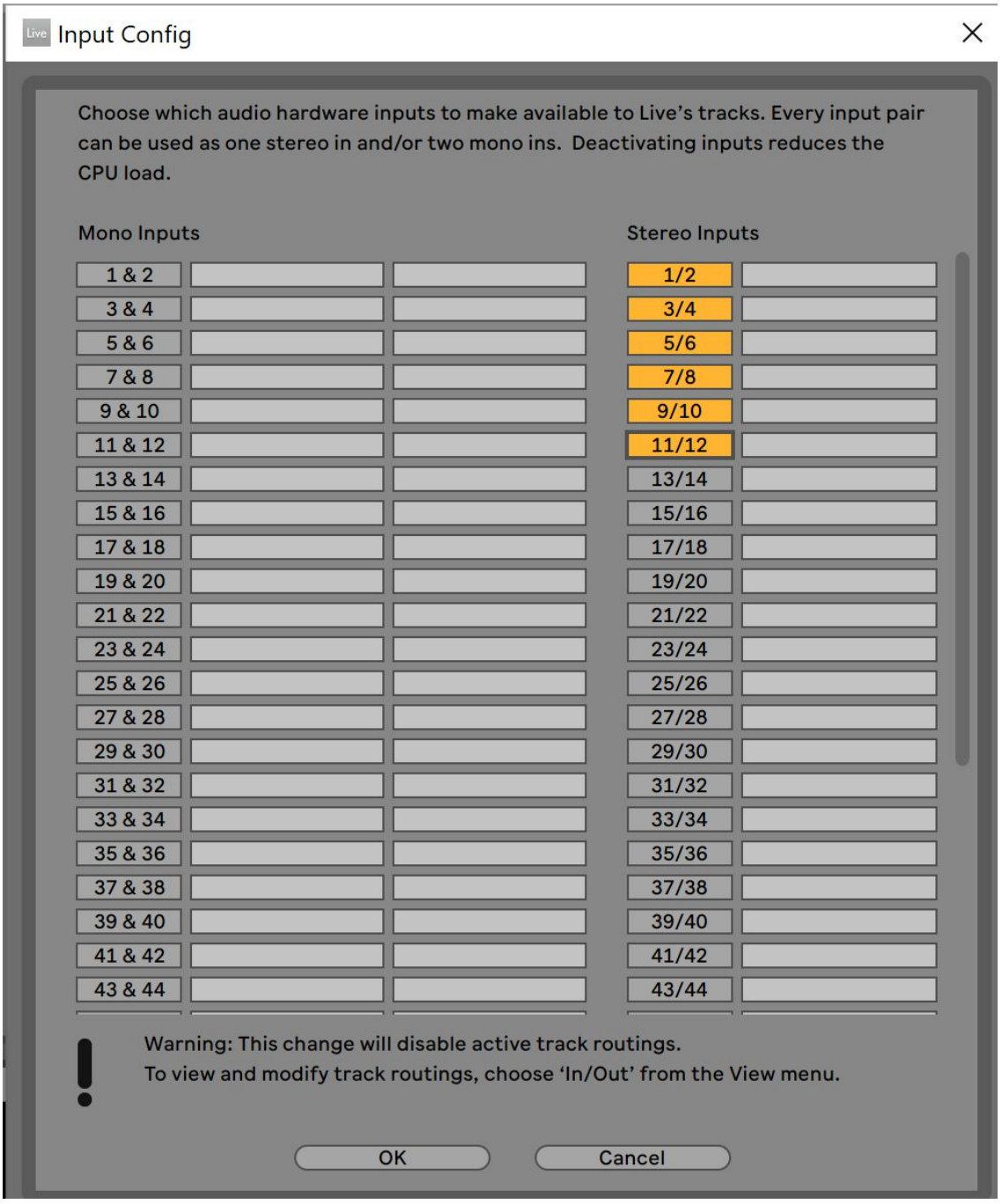
- Make sure jackRouter is registered in the system (chapter 3: Setting up Jack)
- The jack server must be running.



Be advised
To load the plugin properly, the audio driver must be On when ableton starts. Select a driver and reload the ableton project if you get this message:

6.3 In the audio preference menu click: “input config”

> Activate stereo inputs 1 t/m 12

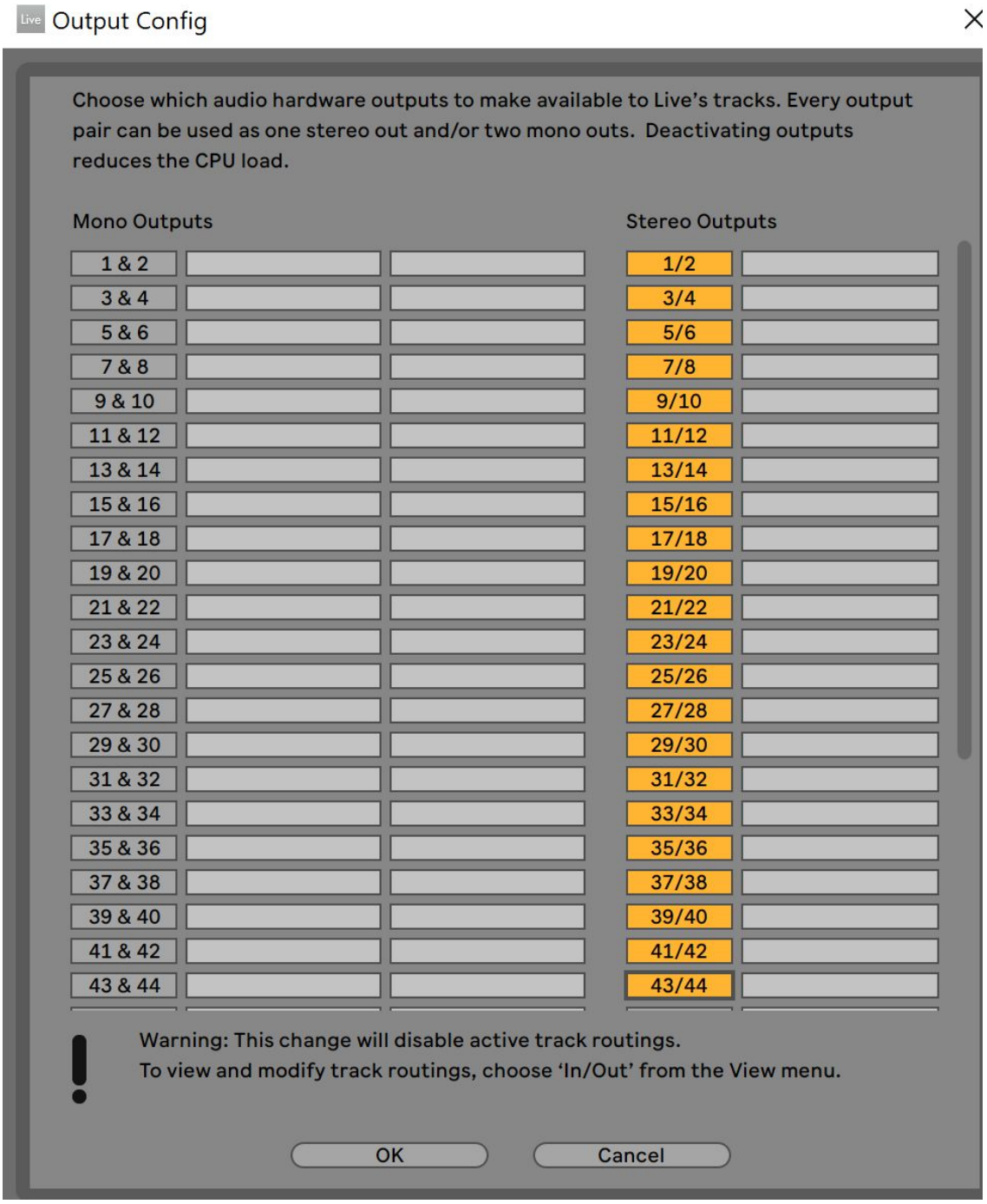


> Select input 1 t/m 12

- Channel 1 t/m 10 are inputs for Spatial Engine
- Channel 11 can be used for a dedicated reverb channel for microphone interaction effects.

6.4 Do the same with the outputs:

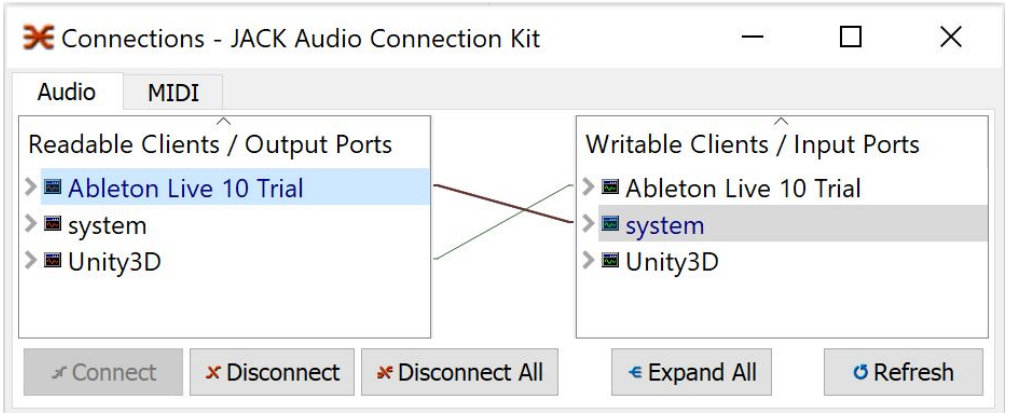
> Click “Output config”
> Activate stereo outputs 1 t/m 44



Ableton will always device audio tracks into stereo pairs.

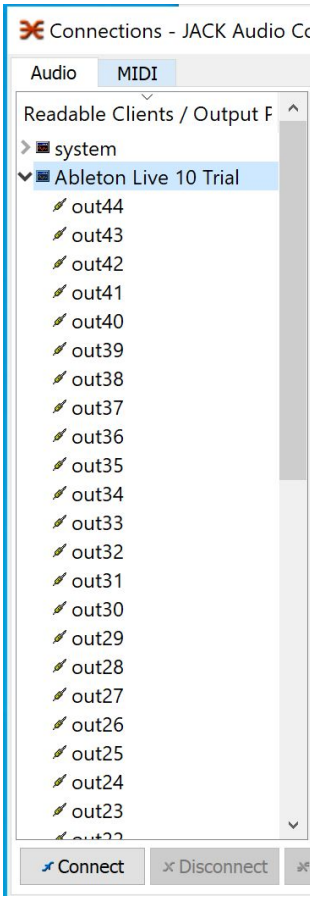
6.5 Unity3D to Ableton live via Jack Server

> Go to Jack Control > Connect
> Connect Unity3D to Ableton Live 10
> Connect Ableton Live 10 to System Output



- Make sure all the IO are showing up

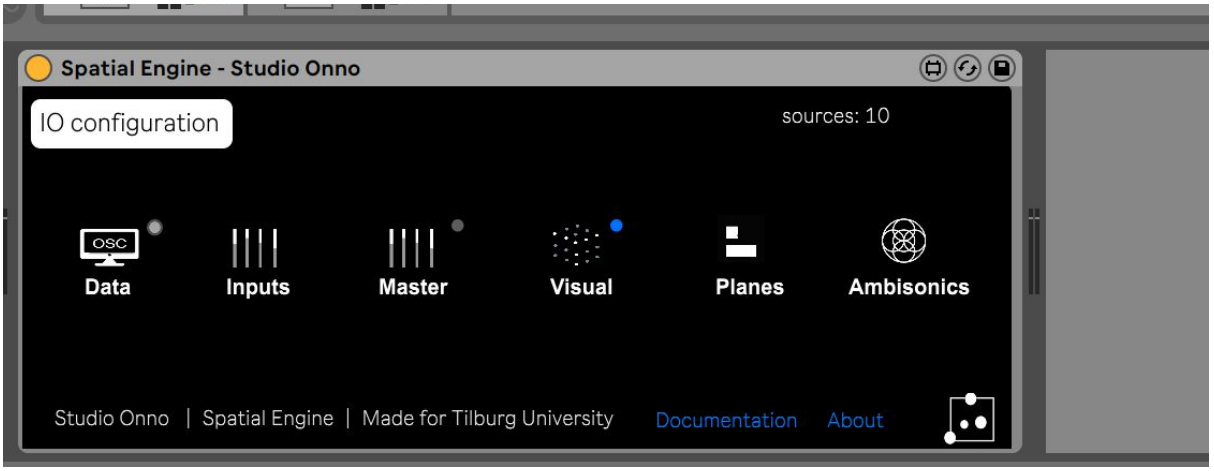
> Make sure the JackRouter.INI file is set to 64 channels (chapter 3: Setting up Jack)

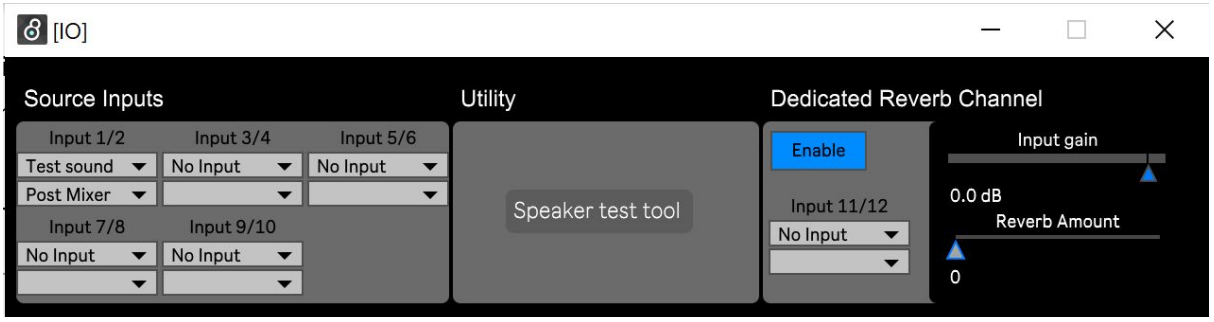


7 Spatial Engine plugin IO Configuration

7.1 Configuring Inputs to Spatial Engine

> In the Spatial Engine Plugin, Click on “IO Configuration”

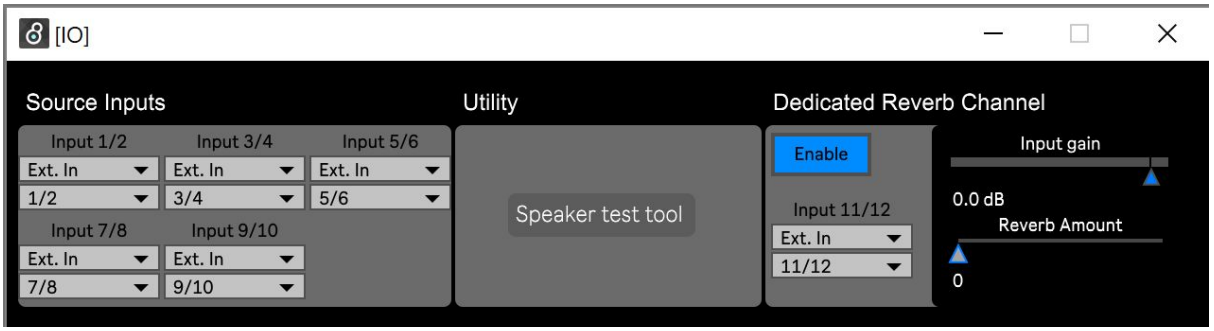




- Setting the inputs to “test Sound, will take the sound from another Ableton channel.
- Setting inputs to “ext. in, will grab sound from jack directly

> For streaming sounds from Unity3D into Spatial Engine: Choose: **Ext. in.**

> input 11 can be routed to a dedicated reverberation channel.



7.2 Configuring Outputs from Spatial Engine to Jack.

> Set all Speakers outputs to Ext. Out, and assign stereo pairs 1/2 - 3/4 - 5/6 etc..

7.3 Headphones modus should be off

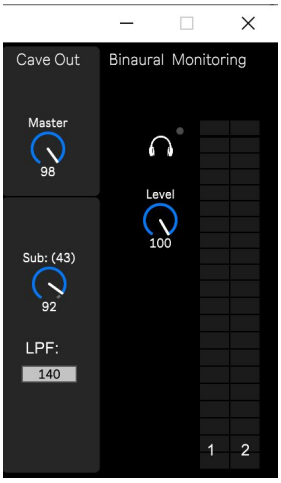
> Navigate to Spatial Engine > master

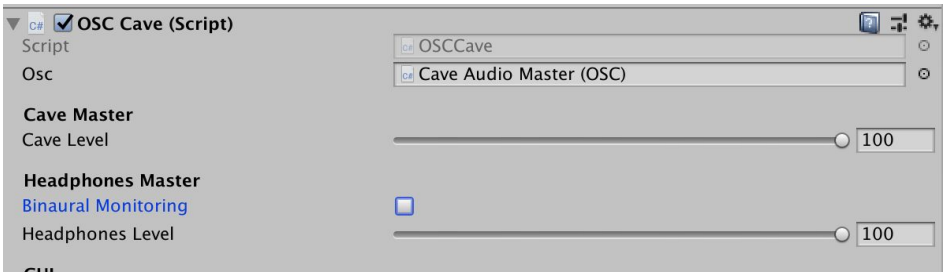
> Turn off binaural monitoring

IMPORTANT! OSC Commands will always overwrite local settings

> In Unity3D - Cave script: Turn off binaural monitoring.

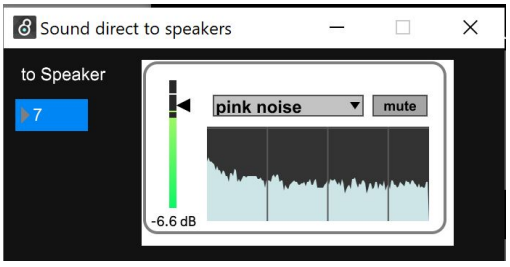
- The same rule applies to all OSC commands.





Headphones mode will downscale 4.2.2 speakers to a binaural stereo signal. When engaged: Audio will come out of the first two outputs.

7.4 Test all the outputs

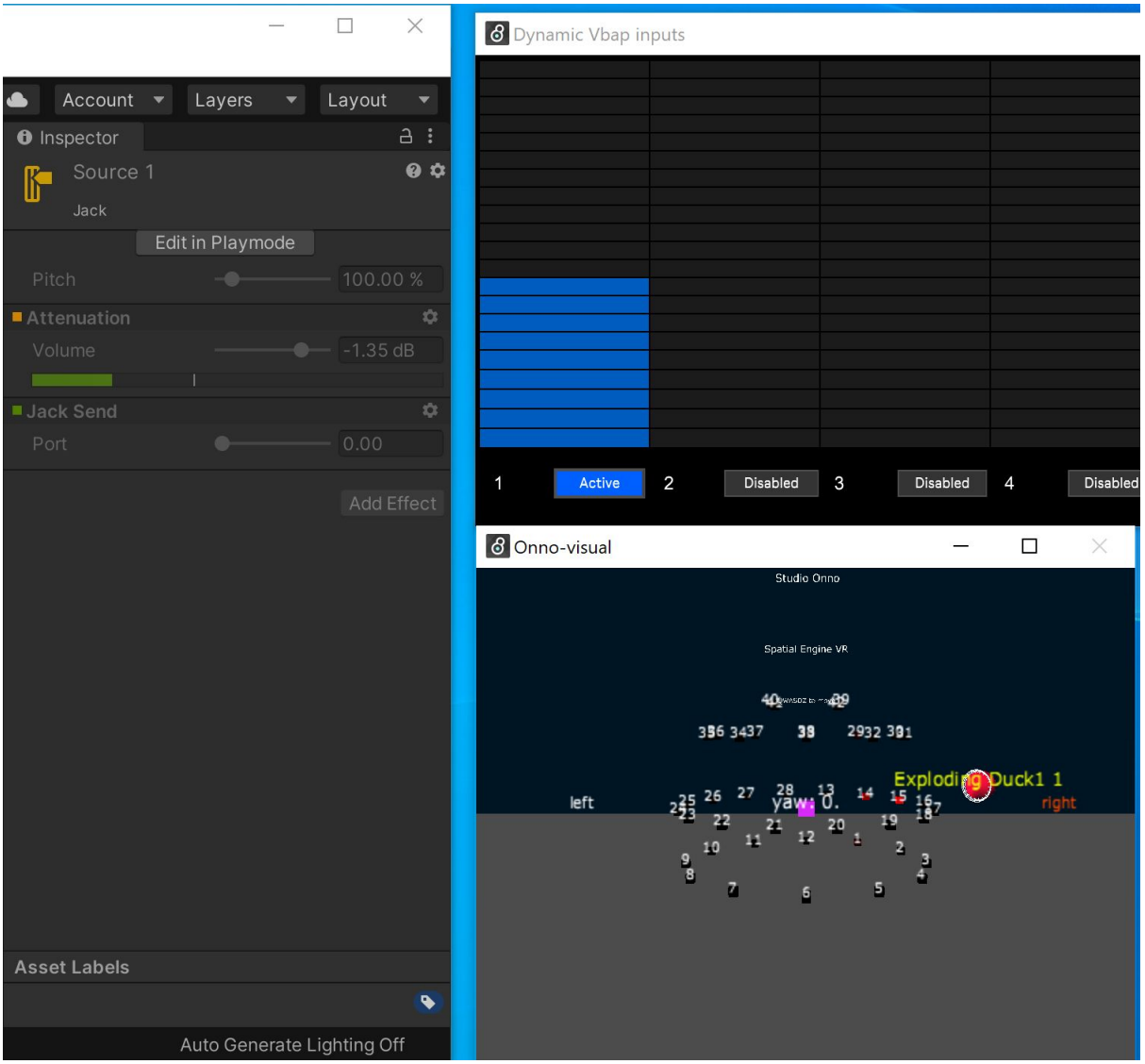


> Use the speaker test tool to check if all the speakers are connected.

8 Using Spatial Engine With Unity3D

8.1 Audio signal from Unity3D to Spatial Engine.
> Run the Unity3D “helloworld” example scene.
> In Spatial Engine plugin: Navigate to “Inputs”

- You should see audio streaming into Spatial Engine
- Make sure the Source in Spatial Engine is set to “active”

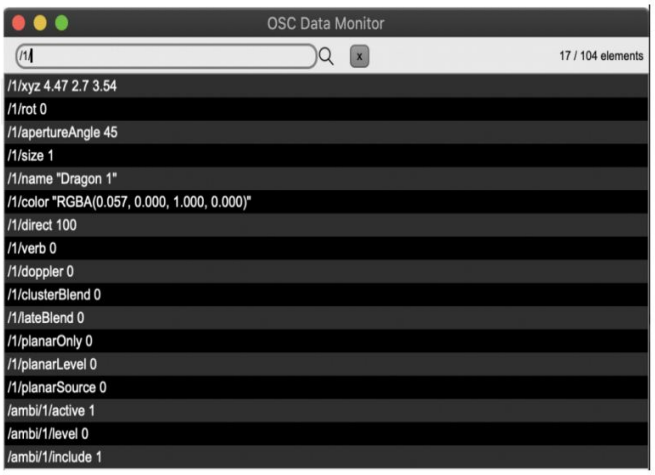


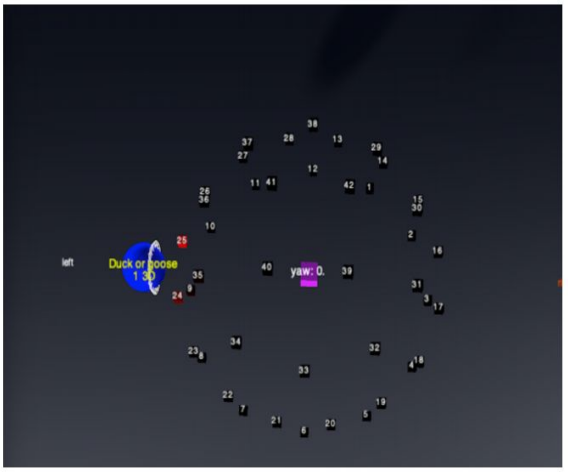
- The volume slider in Unity3D now affects the volume in Spatial Engine
- Up to 64 audio channels can be streamed simultaneously and in realtime
- Latency depends on the selected DSP buffer size and Sample rate in Jack.

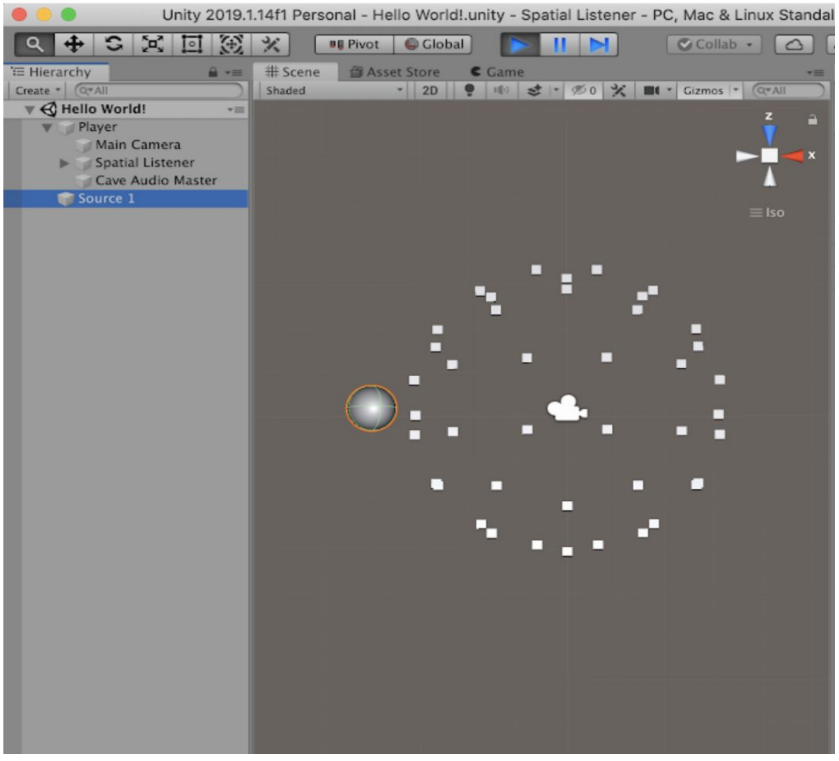
8.2 Moving audio sources around using Spatial Engine.

> Use Unity's transform controls to position, scale, and rotate "Source 1" Inside the Cave.

> **For a full overview of all the controls, please read the OSC Commands list.**







> In spatial Engine, Open the visualizer to see the action

> the Data OSC monitor is used to see all incoming OSC messages in realtime.

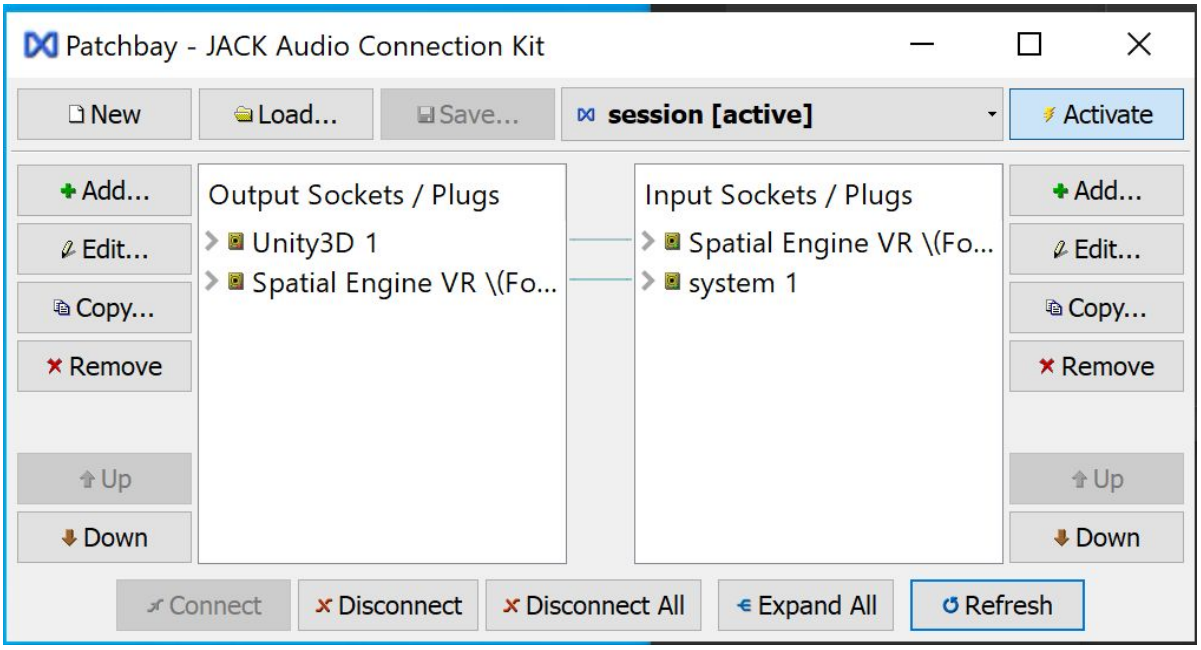
OSC DATA WILL ALWAYS OVERWRITE THE LOCAL CONTROLS!

> OSC Bundle is updated when:

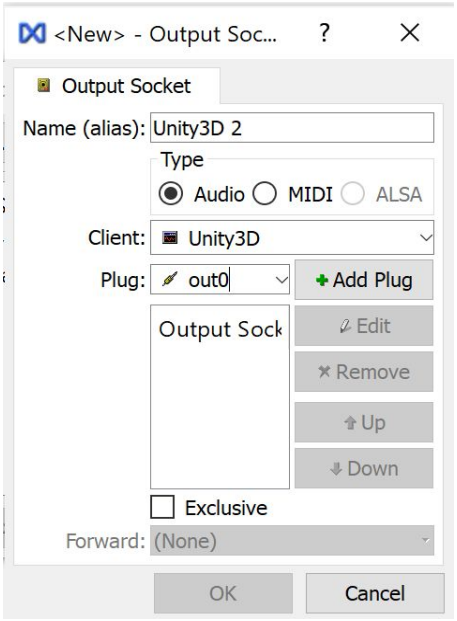
- The game starts: Triggered from the void start function in the Cave script.
- Bundle can be pushed out manually using /osc/push 1 command.
- A parameter value changes

9 Automating connections with Jack

9.1 Adding output sockets
> In jack Control. navigate to *“patchbay > add (left side button)*
> Click “add” in the Output sockets/plugs window.



Here you must add the audio ports manually by clicking “add plug” for the number of ports you want to add. If you are sending 10 channels from unity3D, you should add 10 plugs.



9.2 Adding Output plugs for Unity3D
> Click “add plug” (left side)
> Add 10 output plugs for Unity3D

9.3 Adding Output plugs for Ableton Live
> Click “add plug” (left side)
> Add 44 output plugs for Ableton Live 10

9.4 Add input plugs for Ableton And Unity3D
> Repeat step 9.2. and 9.3 for the Input sockets (right side)

9.5 Activate automatic connections
> In the patchbay: Make desired IO Connections.
> Click “activate (top right corner” To remember the IO.

10 Troubleshooting

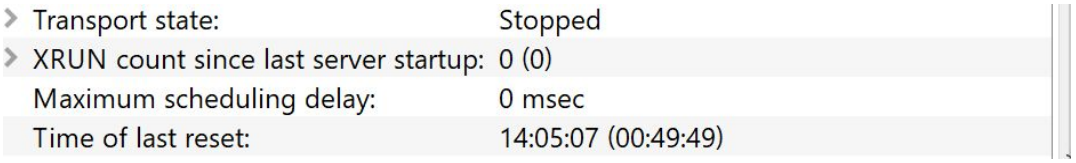
9.1 No audio?
> Make sure everything is connected in Jack as instructed
> Unity3D Sample Rate and buffer size must align with Jack's settings
> Make Sure the Audio source is not on "mute"
> The Digigram Asio LXIP driver should be selected/targeted in Jack.
> The speakers should be turned on.

9.2 Distorted audio?
Contrary to Unity3D's audio mixer, which lets you mix up to 20db+. Jack Audio will clip the signal on 0db, resulting in audible distortion. It is therefore important not to send any sound louder than 0db from the UNity3D mixer. Make sure no audio meters turn red!

9.3 Ableton CPU overload
Make sure the CPU level is OK. Ableton has a CPU meter in the top-right corner. if this percentage becomes very high, audio dropouts will occur. In order to guarantee a stable performance, CPU usage must not exceed 40% due to possible spikes.

- Headphone modus increases CPU usage due to virtualisation
- A lower buffer size increases the CPU usage
- A higher sample rate increases CPU usage
- Sending many OSC commands increases CPU loads.

9.4 Xrun
Make sure the number of "Xruns:" in Jack remains 0. You can see the Xruns under the "messages > Status" tab in Jack Control. Xruns result in audio crackles.



9.5 Unity3D Debugging
> Make sure there are **No duplicate ports for the JackSend plugin!** This will result in crackling audio due to interfering digital signals.
> Do not make changes to jack send plugin while the game is running
> Bottom Up ASLR must be disabled! otherwise Windows defender will encrypt inter-app communication. This leads to jack not being able to receive audio. Unity3D will crash because of a Libjack.dll error.

11 References

Studio Onno Homepage - Spatial Audio research for Tilburg University
https://www.cinjee.com/spatial-audio-research

Resources used

Jack Audio Connection Kit
https://jackaudio.org/

Open Sound Control
http://opensoundcontrol.org/

Ableton Live
https://www.ableton.com/en/live/?gclid=CjwKCAiAm-2BBhANEiwAe7eyFPiKoW63iysF3BbjIUVJp_SrO3h7A76RomZ04Qyubbauk3bn5X-zXhoCa6wQAvD_BwE

Jack for Unity3D plugin
https://github.com/rodrigodzf/Jack-Audio-For-Unity

Ircam Spatial Audio research institute Paris
https://forum.ircam.fr/

Genelec GLM calibration
https://www.genelec.com/glm

Hardware used

Genelec P-smart POE Loudspeakers
https://www.genelec.com/smart-ip

Methods

Ville Pullki - Vector based Amplitude panning
http://lib.tkk.fi/Diss/2001/isbn9512255324/article1.pdf

Ambisonic rendering
https://www.researchgate.net/publication/280010078_Introduction_to_Ambisonics

New OSC implementation of Spat5 for MaxMSP
https://hal.archives-ouvertes.fr/hal-02094499/document

Special Thanks to Thibaut Carpentier for all the support.