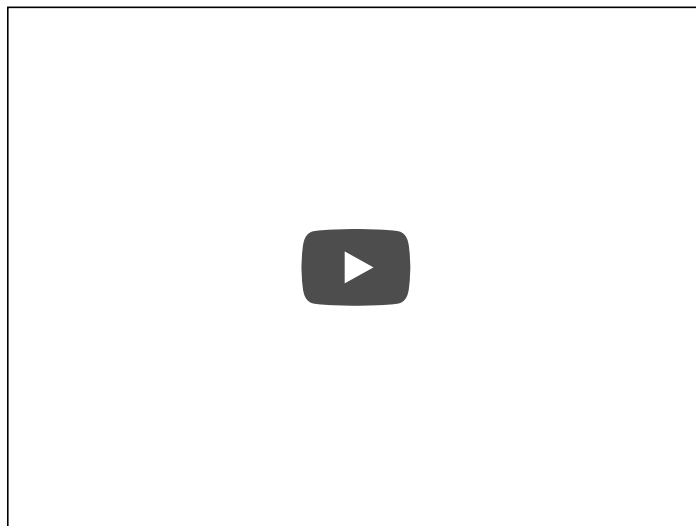
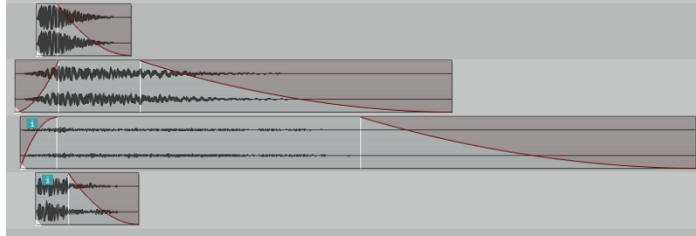


Sound Design (UI)



Christian Grewell [Follow](#)

Oct 15, 2018 · 6 min read



Description

The purpose of the ears is to point the eyes.

In this template, we will be learning about the importance of spatial, or 3D audio for delivering a more immersive experience in mixed, augmented and virtual reality. Spatial audio is important not just for giving our users a sense of presence, but can also provide utility in a number of applications. Think about any situation where location and sound are important. Adding 3D sound to your experience is not a difficult exercise, but is all too often not prioritized over visual elements. Hopefully after completing this, you'll agree on its absolute necessity in contributing to immersion.

Our method of creating soundscapes in this template will initially NOT respond to motion (though we will discuss how to change this using the new sensors at our disposal).

What is Sound?

The sounds we hear are fluctuations in air pressure, small variations from normal atmospheric pressure caused by vibrating objects.

As an object moves, it displaces air molecules next to it, which in turn displace air molecules next to them, and so on, resulting in a momentary 'high pressure front' that travels away from the moving object and towards your ears. If we cause an object to vibrate, such as a string on a guitar, and then measure the air pressure at some nearby point with a microphone, the microphone will detect a slight rise in air pressure as the 'high pressure front' moves by. In the case of a string, because they are fixed at each end, there is a restoring force pulling them back to their normal position, and because this restoring force gives it momentum it overshoots its normal position, moves to the opposite extreme position, and continues vibrating back and forth in this manner until it eventually loses momentum and comes to rest in its normal position. As a result, our microphone detects a rise in pressure, followed by a drop in pressure, followed by a rise in pressure, and so on, corresponding to the back and forth vibrations of the string.

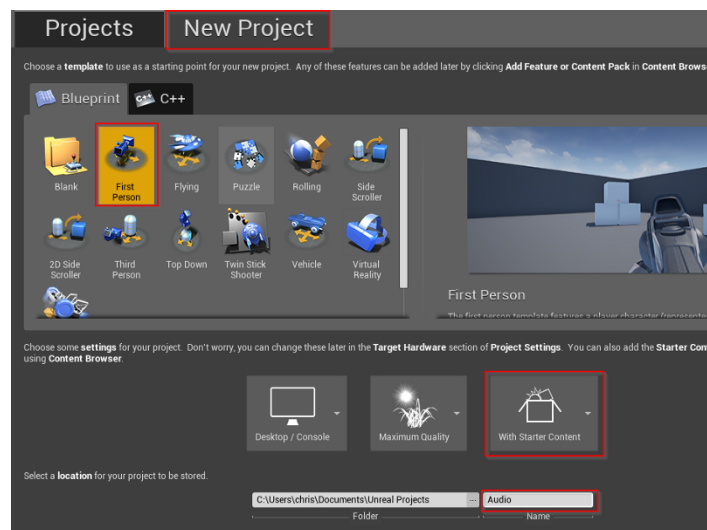
In order for us to hear such fluctuations of pressure:

- The fluctuations must be large enough to affect our eardrum, yet not so large as to hurt us. In practice, the intensity of the changes in air pressure must be greater than about 10–9 times atmospheric pressure, but not greater than about 10–3 times atmospheric pressure. Practically speaking, the softest sound we can hear has about one millionth the intensity of the loudest sound we can bear.
- The fluctuations must repeat at a regular rate fast enough for us to perceive them as a sound (rather than as individual events), yet not so fast that it exceeds our ability to hear it. Usually this range of audible frequencies is 20 to 20,000 cycles per second (abbreviated Hz).

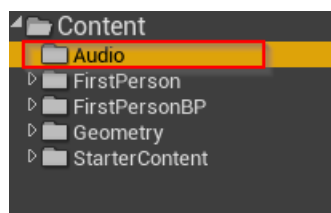
Basic Spatial Audio in UE4



First, let's start a default First Person template project, this will help with understanding how our sounds might be rendered as someone moves about our VR world, later on, I'll show you how to add in the VR content pak and swap out the first person pawn for a VR pawn.



1. Create a new folder called 'Audio'—this is where we will put our sounds and spatialization actors



Now, let's add some audio and configure it for spatialization. Here are a few great resources to download game audio.

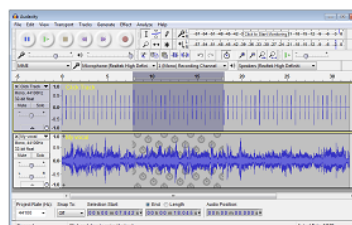
- www.freesound.org
- <https://opengameart.org/>

Download Audacity (<http://www.audacityteam.org/>), it's an open-source sound editing and mixing tool

Welcome to Audacity

Audacity® is free, open source, cross-platform audio software for multi-track recording and editing.

Audacity is available for [Windows®](#), [Mac®](#), [GNU/Linux®](#) and other operating systems. Check our [feature list](#), [Wiki](#) and [Forum](#).



[Download Audacity 2.1.3](http://www.audacityteam.org/)

Mac 474 2017 Audacity 2.1.3 Download

<http://www.audacityteam.org/>

Now it is time to plan out your 3D soundscape. In my example, I'd like to have you sitting in a cozy room while someone enters, closes the door and recites klingon poetry in your ear.

To do this, I need to think in temporal terms and imagine my user experience:

- 5 seconds of crackling fire
- Door opens
- Dootsteps (2 secs)
- Door closes
- Door locks

Our Files:

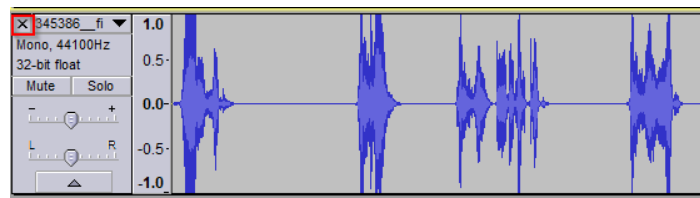
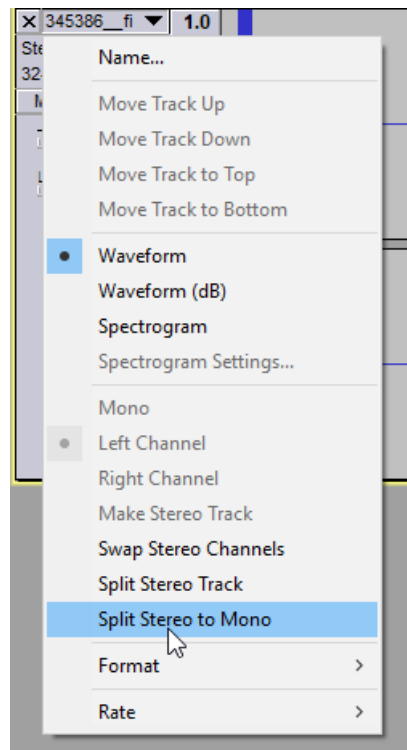
— **Fire:** <http://freesound.org/people/kingsrow/sounds/181563/>

— **Door Open / Close:**
<http://freesound.org/people/amholma/sounds/344360/>

— **Klingon Poetry:**
<http://freesound.org/people/fiatknox/sounds/345386/>

Spatial sounds should be **mono** (not stereo). So let's go into Audacity and make sure they are prepared correctly before importing them into UE4.

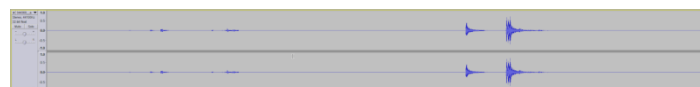
1. Open the sound in Audacity
2. Open the audio file you downloaded
3. Click on the drop down and select **split stereo to mono**
4. Delete one of the track



Press the **Spacebar** to sample the sound.

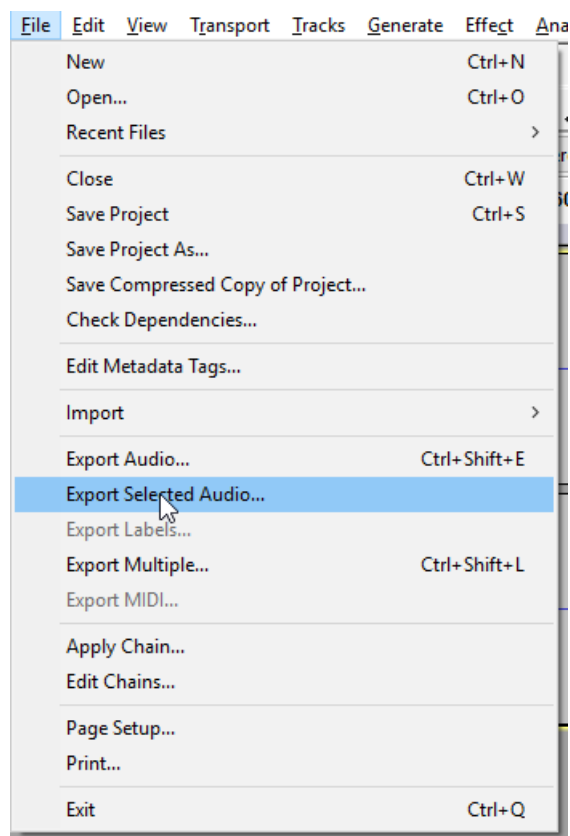
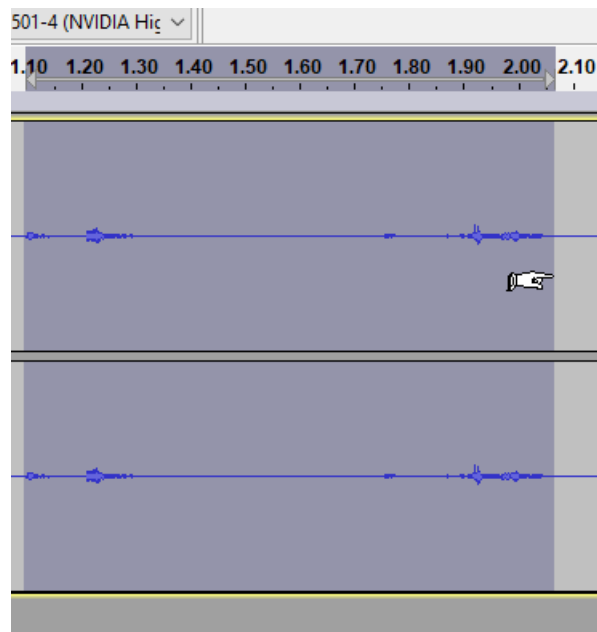
Be sure and remove any silence at the beginning of the sound, by click-dragging the area you wish to remove and pressing delete.

In the case of the door sound, we have two sounds, we'd like to split these into two different files



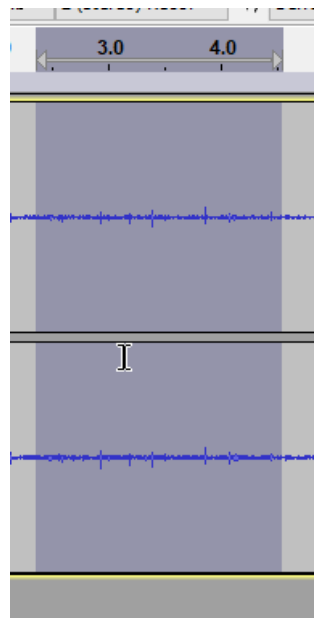
Click and drag on the area you'd like to save as a new file, then press **spacebar** to sample it.

Once you're happy with the sound, go to File > Export Selected Audio and export it.



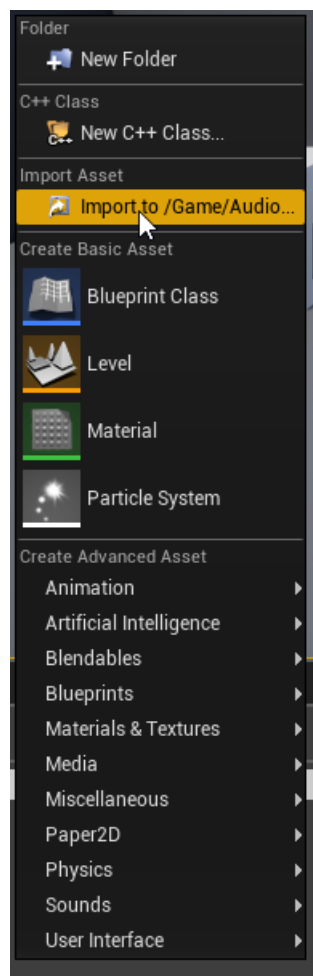
We'd also like to have a looping fire sound.

To sample what a sound would sound like when looped, **hold the shift key and press spacebar**

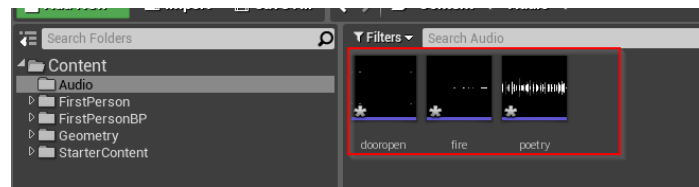


Importing in Unreal Engine

Back in UE4, open the Audio folder you created, then right click in the content browser window and select **Import to**

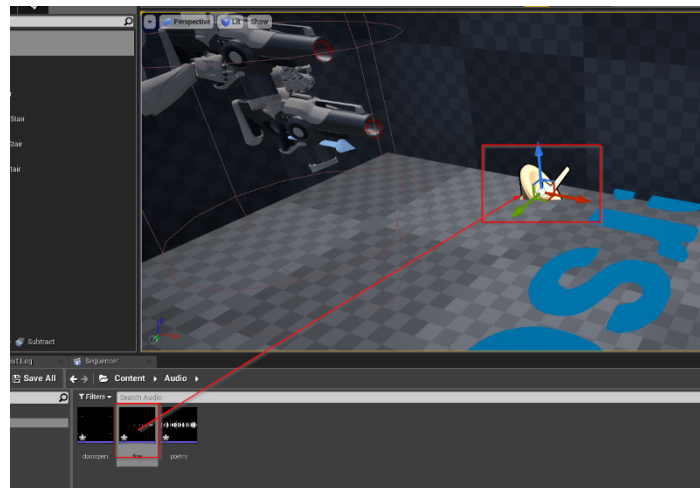


Select all of your prepared audio files and import them. You should now see them in your content browser.



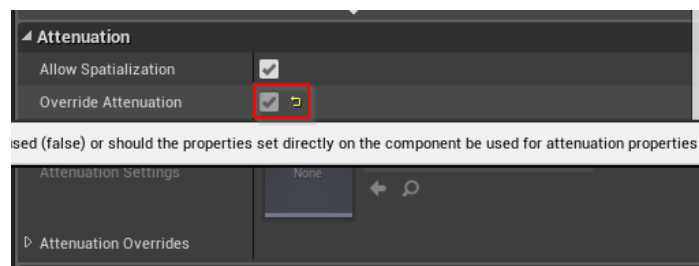
One rapid way to get your sound in the level is to just drag-and-drop it into your scene from the content browser.

This works well for non-interactive sounds.



The default attenuation settings are not going to be very good for this particular sound (e.g., it doesn't naturally decay in volume as we move away).

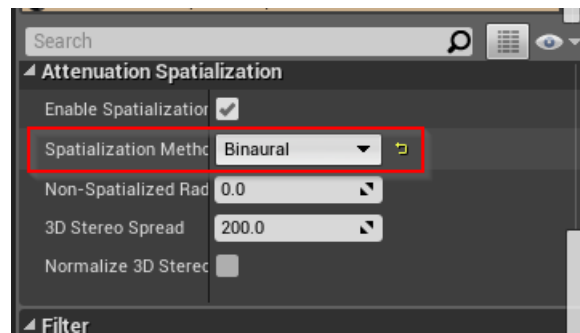
Check the Override Attenuation box in the details panel of your sound



The biggest boost to presence you can do is to use the HRTF spatialization algorithm.

Expand the Attenuation Overrides menu

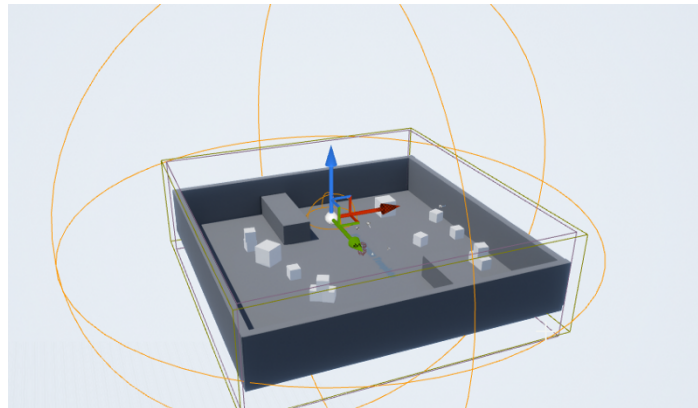
- Select **Binaural**



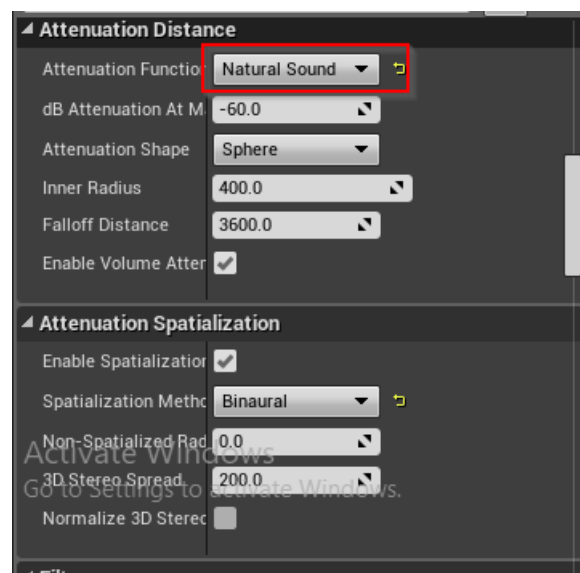
Now, press PLAY and try out the new HRTF spatialization, you'll already notice a huge difference.

If you select the audio, you'll see two spheres, these correspond to the Attenuation radii, the smaller sphere is the radius within which the sound is at its loudest volume, and the outer sphere is where the sound begins to attenuate (get louder)

Outside the sphere, the sound is no longer audible.



For natural sounds, I like to select a non-linear distance algorithm. Play around with these to see what makes sense for the soundscape you are making



This will already sound **REALLY GOOD**. But it can be better.

We want to now use Steam Audio to allow us to occlude (mask out) sounds based on where the player is in the space.

You can enable this by selecting **Enable Occlusion** in the details panel

