Ambience Implementation (Feb 2022)

Created By	
Stakeholders	
Status	
Type	
Created	@February 15, 2022 10:18 PM
Last Edited Time	@February 17, 2022 5:04 PM
Last Edited By	

Outline

This document will outline the updated procedures to implementing ambience in a Ready or Not level. As many of the puzzle pieces are already in place both in Unreal Engine and FMOD, this makes it relatively easy to quickly setup audio within a level. (As long as you already have some sounds done up already!)

Before proceeding, it’s advised to do some reading on FMOD command instruments to better understand how we’ll be using these to change ambiences within a level’s single ambience event. Some FMOD know-how is also recommended as well.

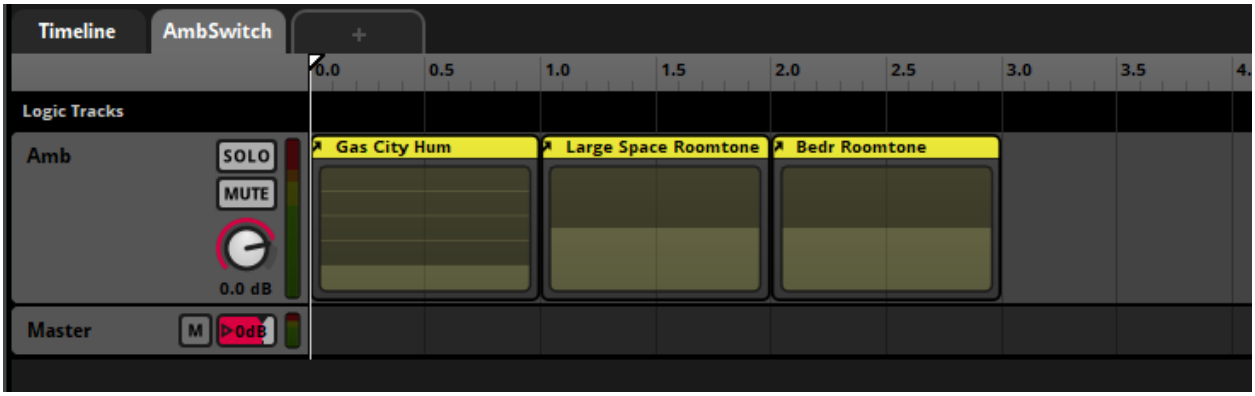
Setup - FMOD (2.02.03)

First thing’s first - we need a 2D FMOD timeline that holds all of the ambiences within a level. Create a new 2D timeline and place it into the corresponding level’s folder in FMOD (e.g. ‘Gas_Amb’). Be sure to assign the new event to the correct level bank.

Create a new loop on the timeline that is two seconds long (the default). Create a new parameter sheet using ‘AmbSwitch’ as the parameter.

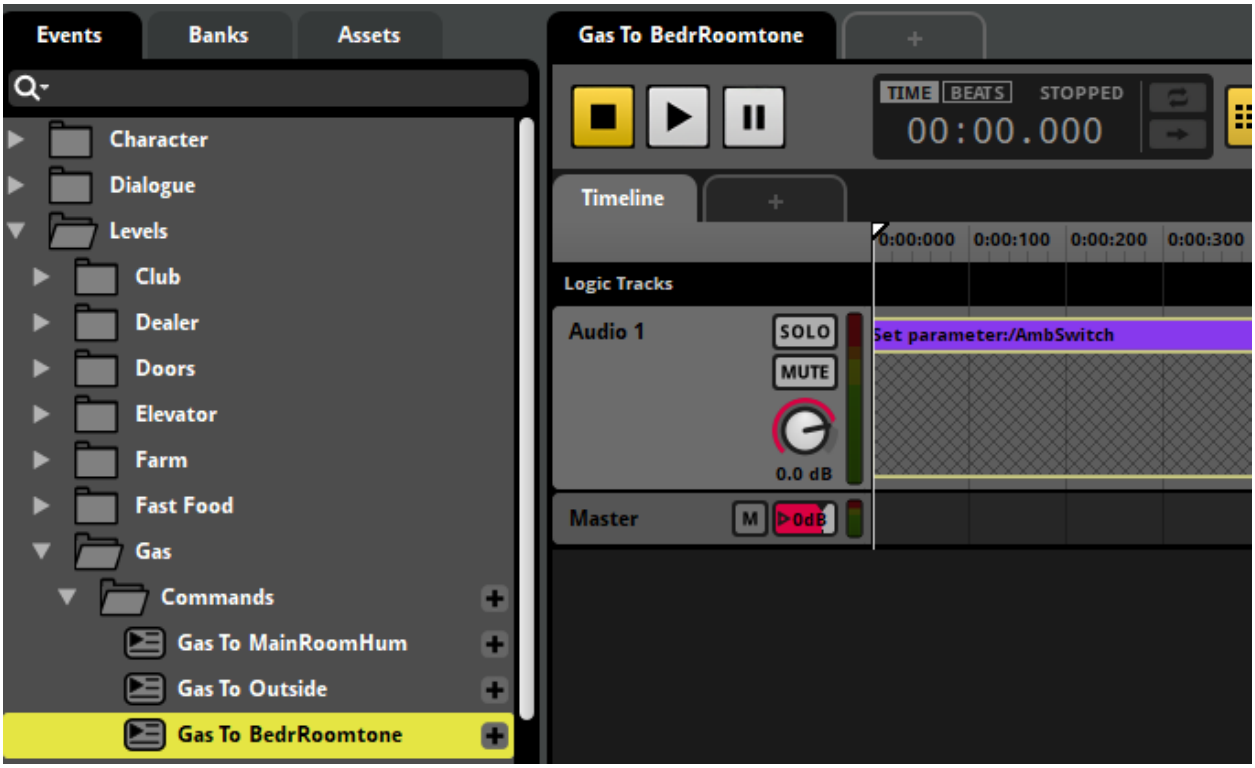
Next, with the AmbSwitch tab open, drag every ambience the level will be using into the timeline space. (These appear on the left hand side under the ‘events’ tab). Drag the event so it is one integer long. As a rule of thumb, using the ‘main’ ambience first, followed by the secondary ones, will make it easier to remember how you’ve set it up later on.

Now if you play the event and change the AmbSwitch parameter to the various integers you’ve set up, the ambiences will change... Neat!

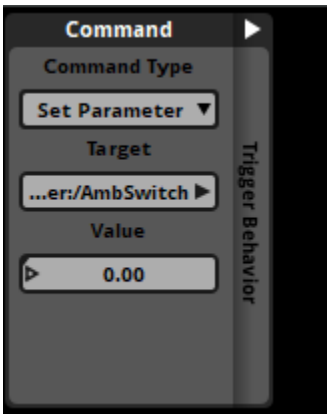


Now we’ll need some command events to tell FMOD to change ambiences.

If there isn't one already, create a new folder called 'Commands' within a level's folder in FMOD. Next, create a new 2D timeline for however many ambiences you have in the main ambience timeline. Gas has 3, so we'll need 3 events. Be sure to name each one so you can easily identify which ambience each event will switch to.



Next, add a command instrument to each event. Change the command type to 'Set parameter', and select 'AmbSwitch' in the menu. Set the parameter to the integer you have assigned for that particular ambience. (So if A is on 0, set the 'Gas To A' command event to 0, etc)



Once all the command events are finished, you'll be ready to implement the ambiences into unreal engine. With the RoN project open in Unreal Engine, build the corresponding level's banks so the new events appear within UE.

Ambience / Reverb Setup (2D / Spatial) - Unreal Engine (4.27)

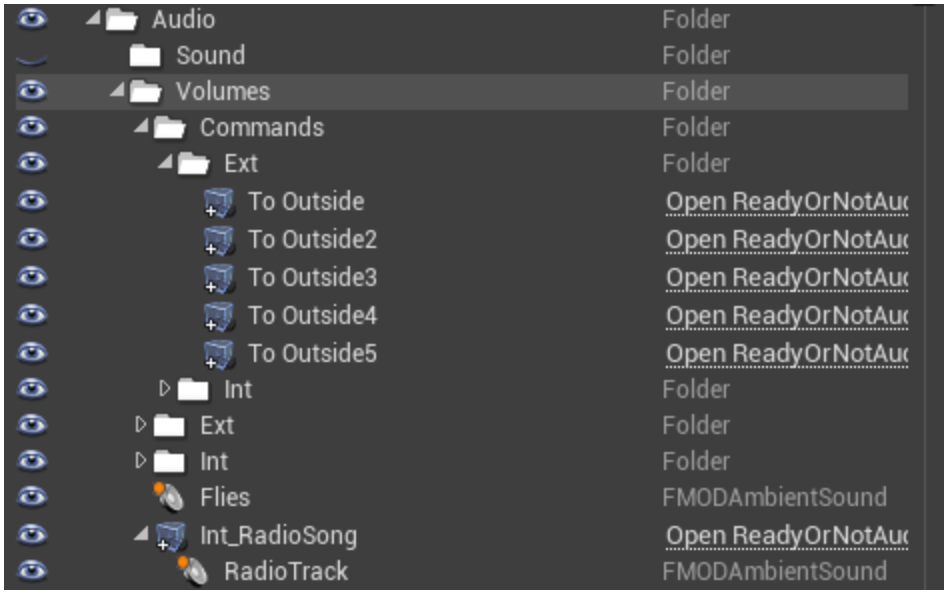
WARNING - BE SURE TO FIRST SELECT THE CORRECT SUBLEVEL BEFORE ADDING ACTORS IN A LEVEL. THIS IS GOOD PRACTICE TO EASILY BREAK DOWN LEVELS INTO EACH LAYER.

For Ready or Not's environmental audio, we mainly use 2 audio actors within a level to flesh the sound out - a 'Ready or Not Audio Volume', and an 'FMOD Ambient Sound'. These are the bread and butter of crafting both the reverb areas and ambient areas, as well as dictating where spatial sounds within a level will play.

The major selling point of our audio volumes is that they're very cheap. We can add plenty to a level without any real impact on performance. There are some caveats, however...

- They cannot be rotated
- They must be a box shape. The box can be scaled on its XYZ axis, but any translations to edges or vertices will “break” the volume.

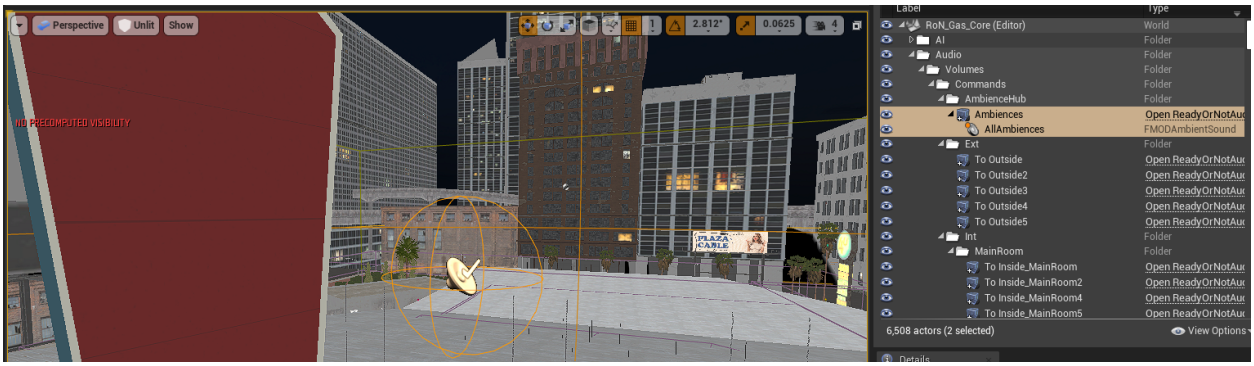
Before creating new audio actors, be sure to create folders within UE to keep things organized.



Ambience / Volumes

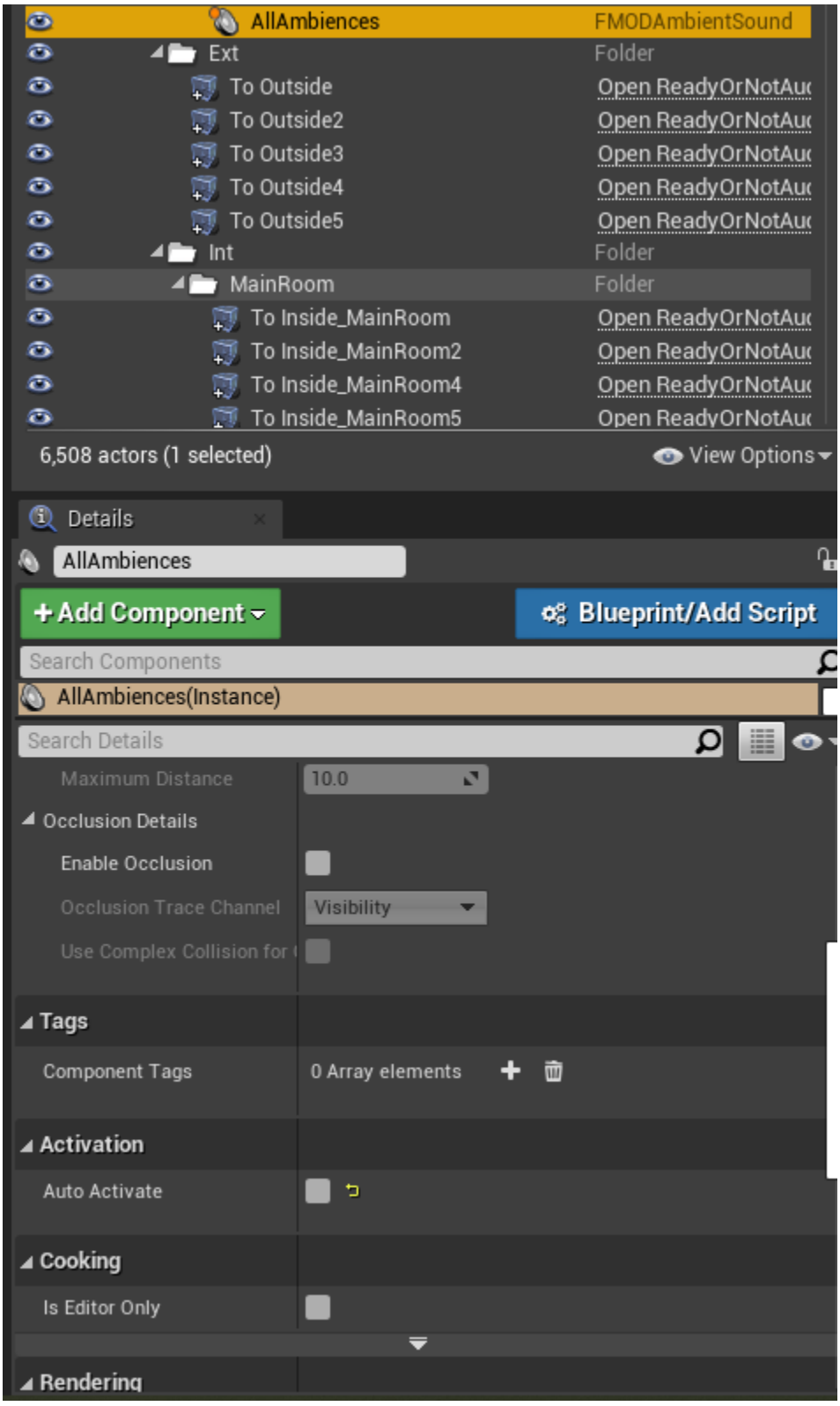
We’ll start by creating the ambience bed for the level.

First, we’ll need a ‘Ready or Not Audio Volume’ that covers the entire level, with an ‘FMOD Ambient Sound’ nested inside. Create a new RoN Audio Volume, and drag the bounds or scale the volume over all the geometry. Attach the new ambient sound you just made to the volume.



Set the ambient sound's event as the main level ambience event. If you named the event 'Gas Amb', it should show up as 'Gas Amb'.

Under the settings for the ambient sound, be sure to untick 'auto-activate'. While not required, it's good practice to turn this off and nest ambient sounds inside volumes to cut down on the audio CPU footprint. We'll get more into that later, though.



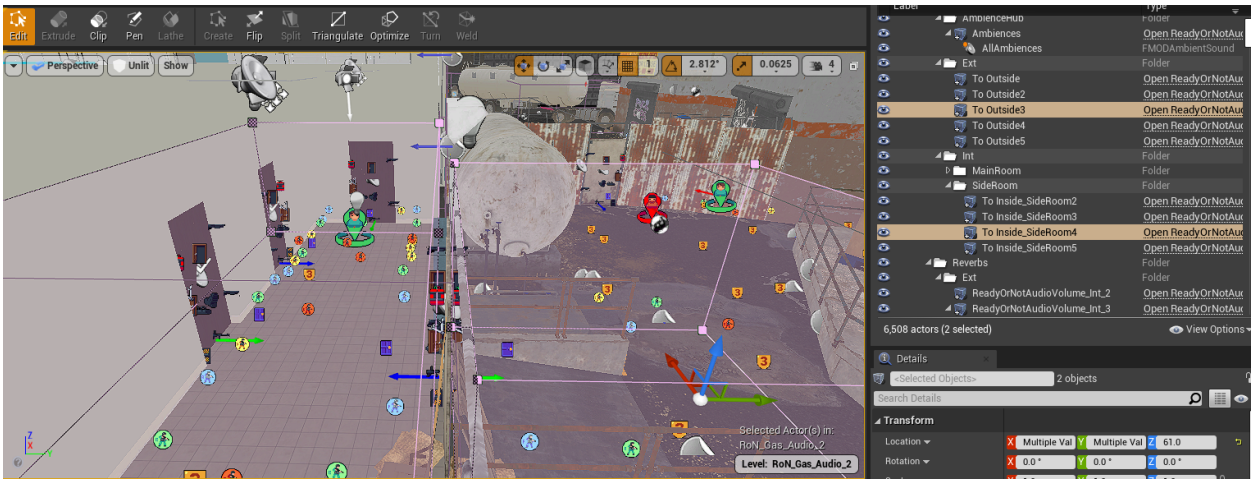
Note that ‘Auto Activate’ is disabled under the ‘Activation’ tab

If you play the level, you should hear the ambience playing - but it won't change as you traverse the environment. This is where those special command events come into play.

Our earlier ambience system required the player to be inside the volume at all times for a specifically selected ambience to play - all we'd have to do is drag the volume over the area we want the ambience to fire, assign the ambience, and voila. Unfortunately for more complex levels with rooms that weren't aligned on a certain axis, this would be a huge problem. As many volumes with the same ambience were required for complex rooms, it would often cause the sound to re-trigger. Farm is probably the best example of this - a huge sprawling map with complex geometry. Setting up cubes that were aligned to each room everywhere would drive anyone insane!

Enter our new system - audio volumes are placed on both sides of a doorway with their corresponding command event assigned to each one.

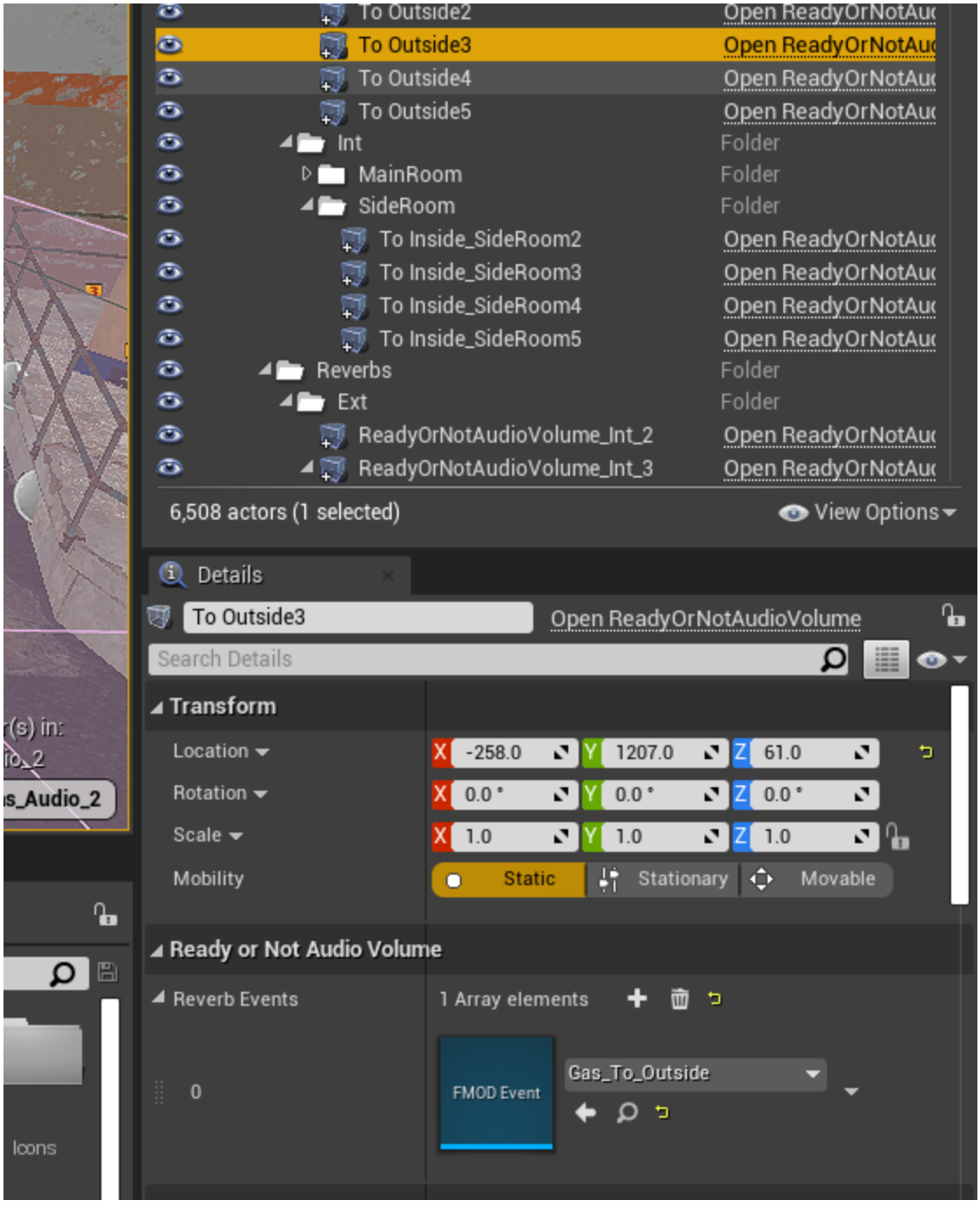
Look at the example below:



Observe how the two volumes are positioned - 'Side room' is placed on the interior (left), and 'Outside' is on the exterior (right). As the player steps into either one, the

ambience will change to the correct ambience.

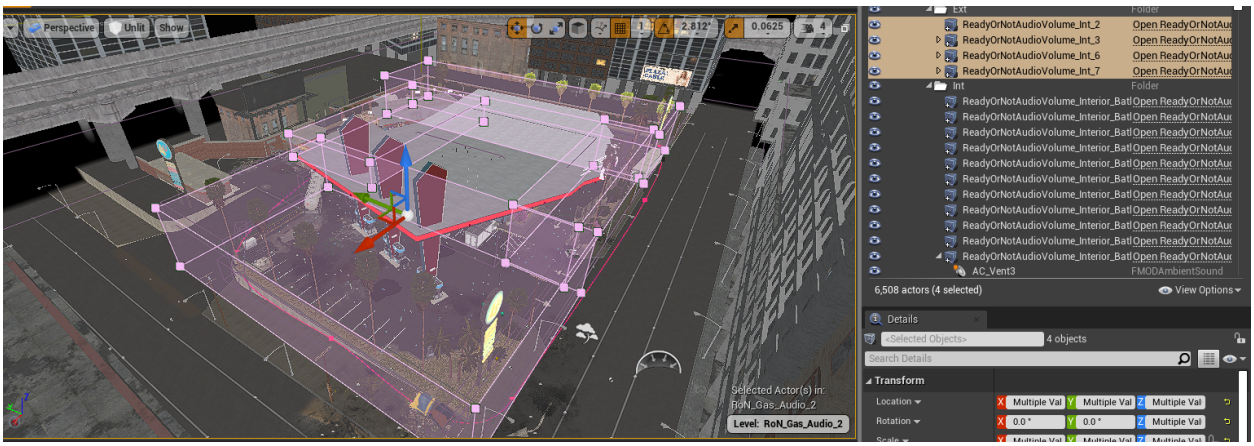
Simply place a volume on both sides of doors where the ambience will change, and assign the correct event using the search function. To assign the event, hit the plus sign under ‘Reverb events’. Don’t worry, you can assign more than just reverbs here!



Keep in mind, this isn’t needed everywhere! Just on doors where you know the sound will transition. Once you place volumes across the location, be sure to test it out to make sure it all sounds great.

Reverb Volumes

To add reverb volumes, the idea is similar to ambience volumes - albeit slightly different. You’ll assign a reverb event to a new volume similar to how you’d assign an ambience to a volume, but the volume must cover the area where the reverb is being applied. When assigning the event, search for reverb and the specific reverb FMOD events will show up. You’ll need multiple volumes for this, but because we use an overriding snapshot instead of an event, it is much less obvious that a transition is happening.



4 reverb volumes are used on the exterior of Gas to cover the external playable area.

It’s also much easier to get away with imprecise volumes for reverbs, as they aren’t as immediately apparent as ambient sounds. (That isn’t to say you should put an exterior reverb inside...)

Spatial Sounds

Now that you've created the "2D" ambience for your level, it's time to start adding in spatial sounds.

Spatial sounds play from a specified location within the level. These actors are called 'FMOD Ambient Sound'. By strategically nesting these within the Ready or Not audio volumes that you've placed around the level, you can significantly lower the audio CPU footprint. The other reason why this is effective is because it gives priority voices (like footsteps and gunshots) more breathing room.

As a good rule of thumb, once you nest the spatial sound, be sure to untick 'auto activate'. This prevents the sound from playing before the player is inside of the specified volume. An example of how we use this in a level is to prevent an interior sound being audible from outside, and vice versa.

Note how the 'Takeout Window' sound is nested in the 'Int_RestaurantTakeoutSound' volume

Getting Creative with Volumes and Outro

You’ve detailed a level, but want to take the sound a step further. Using command instruments, parameters, and volumes, you can create some powerful soundscapes that change dynamically as the player traverses a level.

Need the music to fade to silence when the player enters an area with a stereo playing, and gradually fade back in once they leave? Create a new parameter that a music timeline can use, create some command events, and assign those to volumes around the area. Need more control on how something is occluded (i.e. Club music)? Command events and volumes can help you achieve much more directed occlusion.

That about wraps it up on this short Ready or Not audio guide. The sky really is the limit when it comes to sound design. Don’t be afraid to try something new!