GLOBAL AUDIO MANAGER – DOCUMENTATION

1. Introduction

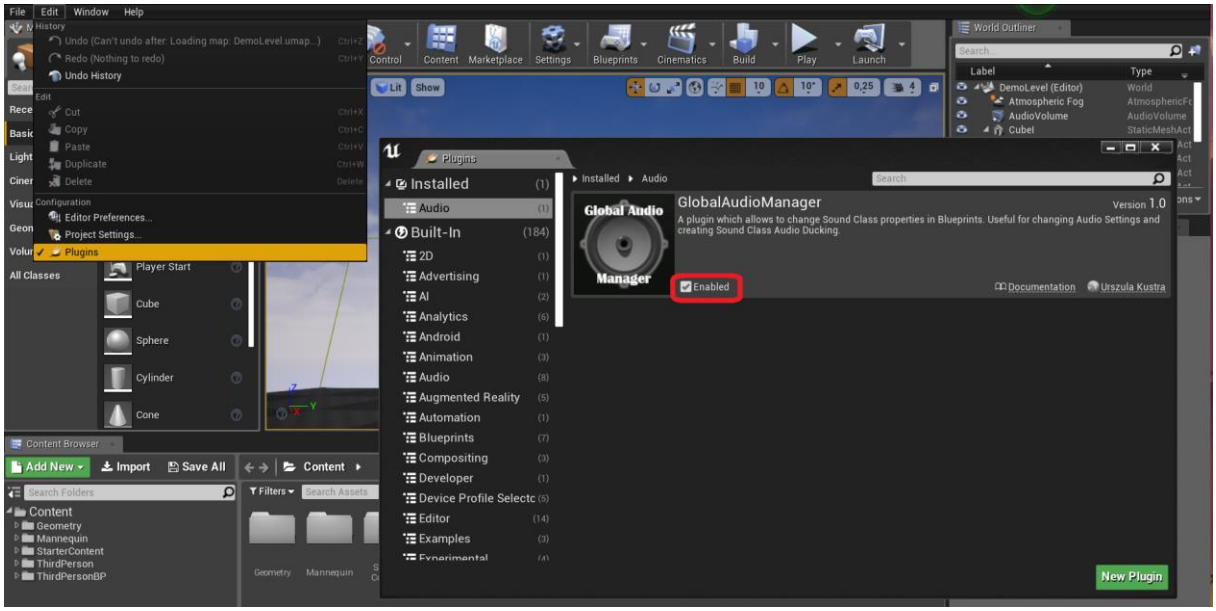
Global Audio Manager (GAM) is a plugin which allows to change Sound Classes properties in Blueprints. Useful for changing Audio Settings and creating Sound Class Audio Ducking.

2. Sound Classes – what they are and what can be done with them by using GAM?

Just a quick reminder: Sound Classes are audio buses – they are assets which stores properties that can be applied to a number of Sound assets. For instance, if you create the "SFX" Sound Class and add it to several Sound Cues/Sound Waves, all of those sound assets will have properties from that Sound Class. So, if you want to change the volume, pitch or reverb of hundreds of SFXes, you can do so by changing the properties of just one Sound Class asset!

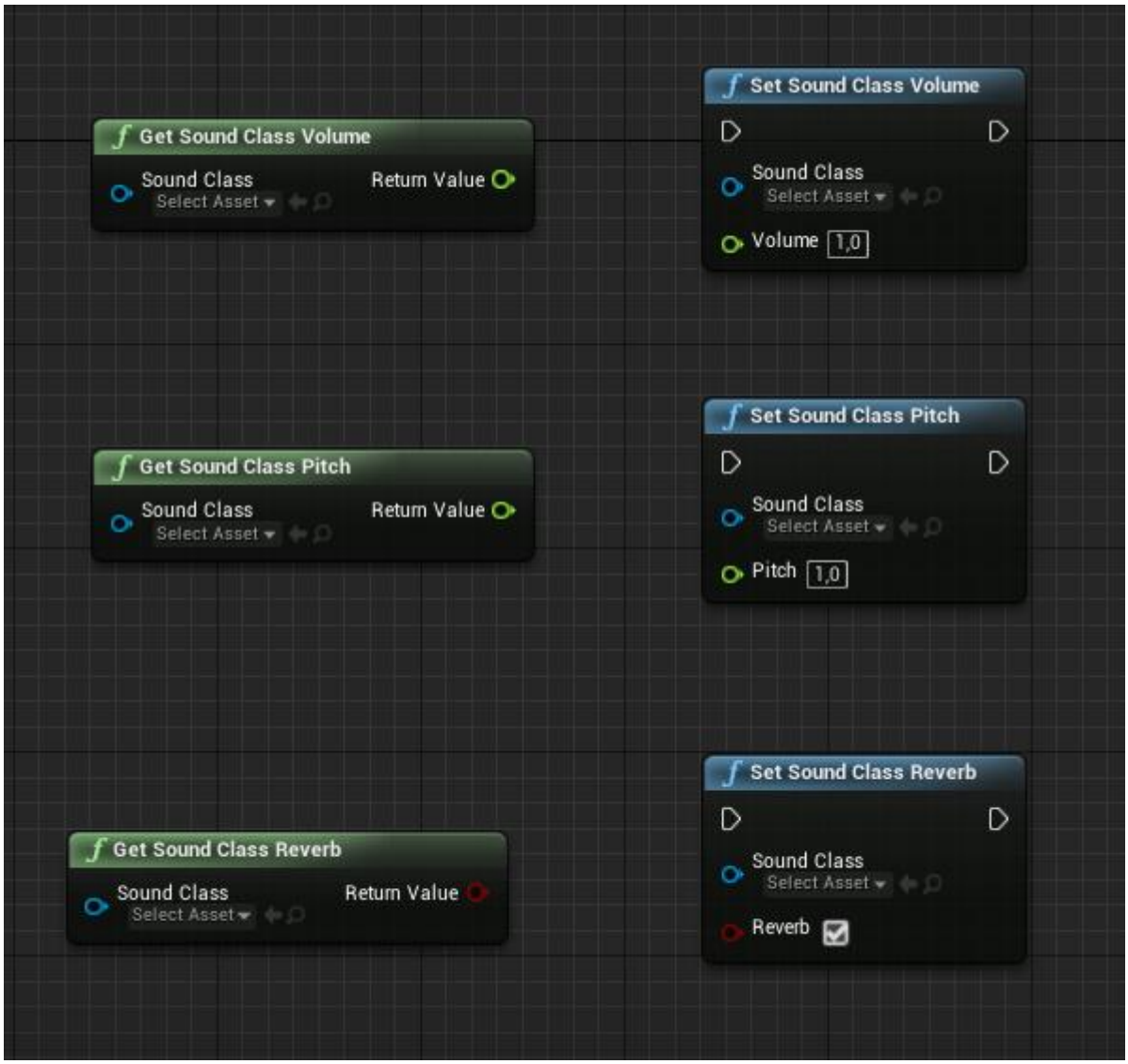
However, changing Sound Class properties in Blueprints is not the only thing this plugin has to offer. It also enables Sound Class Audio Ducking – GAM allows you to smoothly fade out/in some Sound Classes volumes and leave other Sound Classes with an unchanged volume level.

Before proceeding any further, please go to **"Edit->Plugins->Audio"** in the UE4 editor and make sure that Global Audio Manager is enabled:

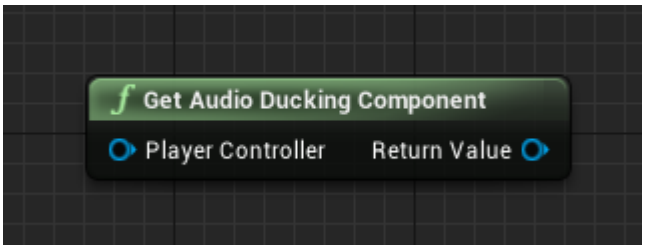


3. New static functions provided by GAM

Functions connected with Sound Class properties:

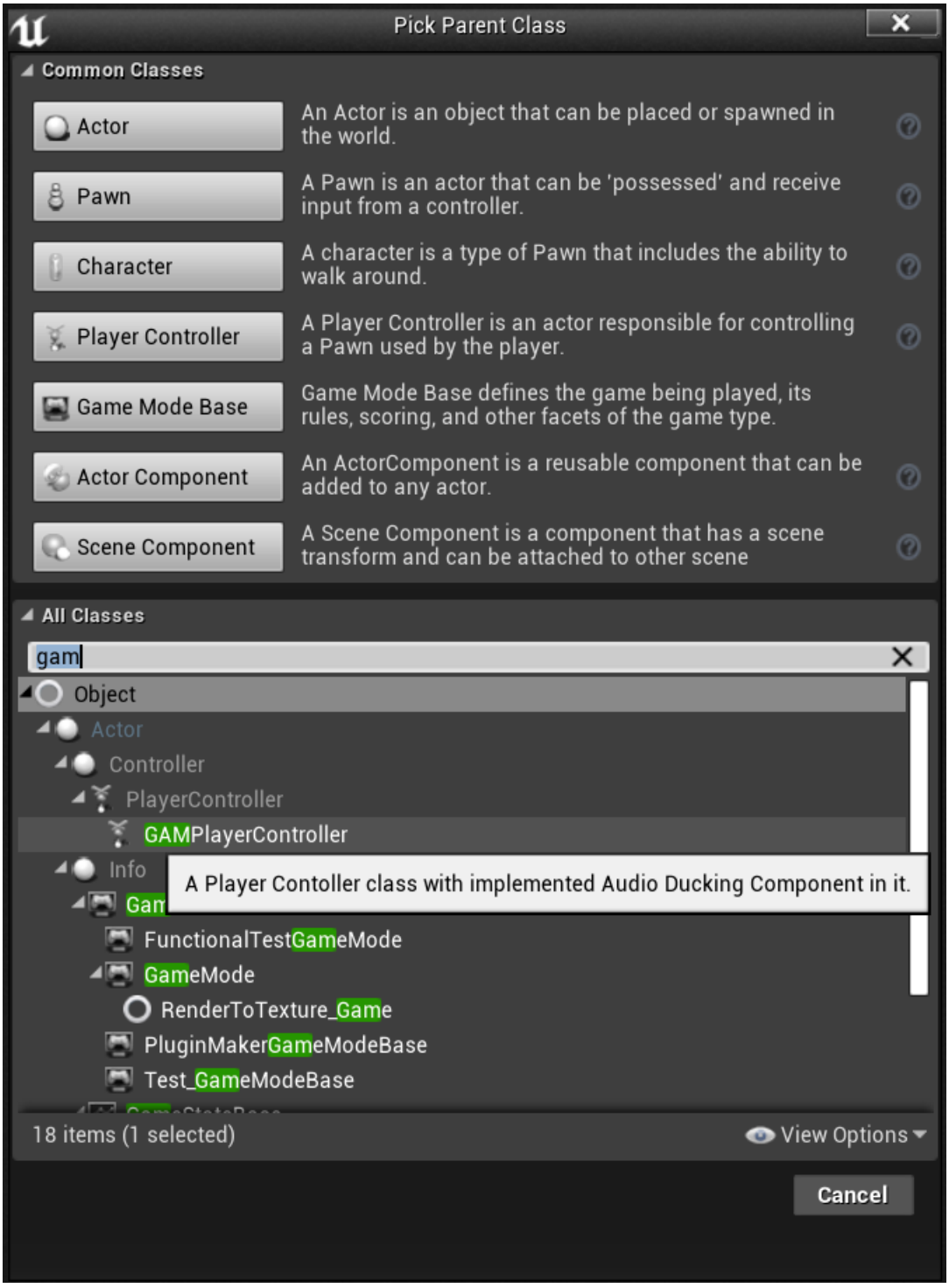


A function which returns Audio Ducking Component from a Player Controller, which inherits from GAMPlayerController class:

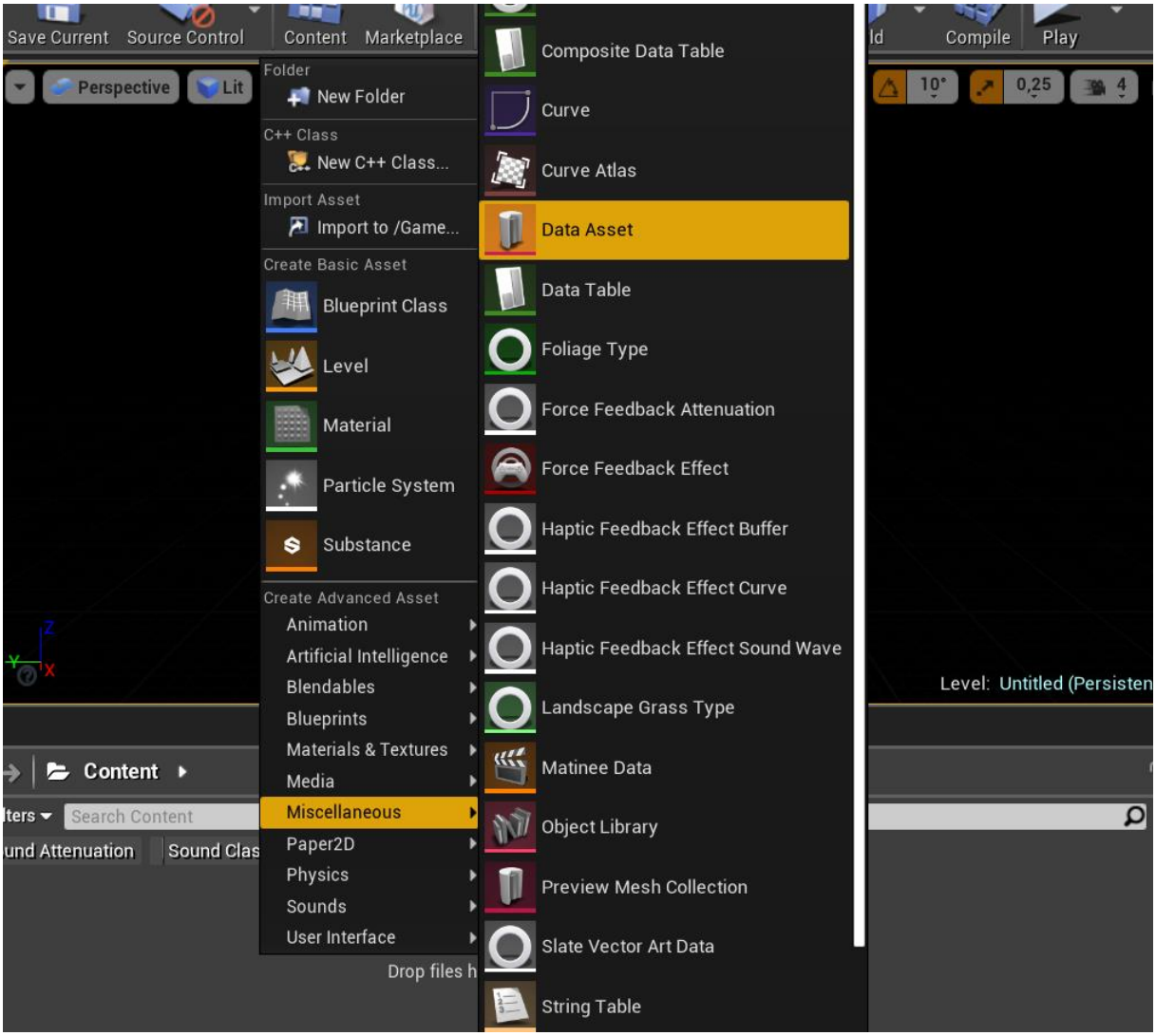


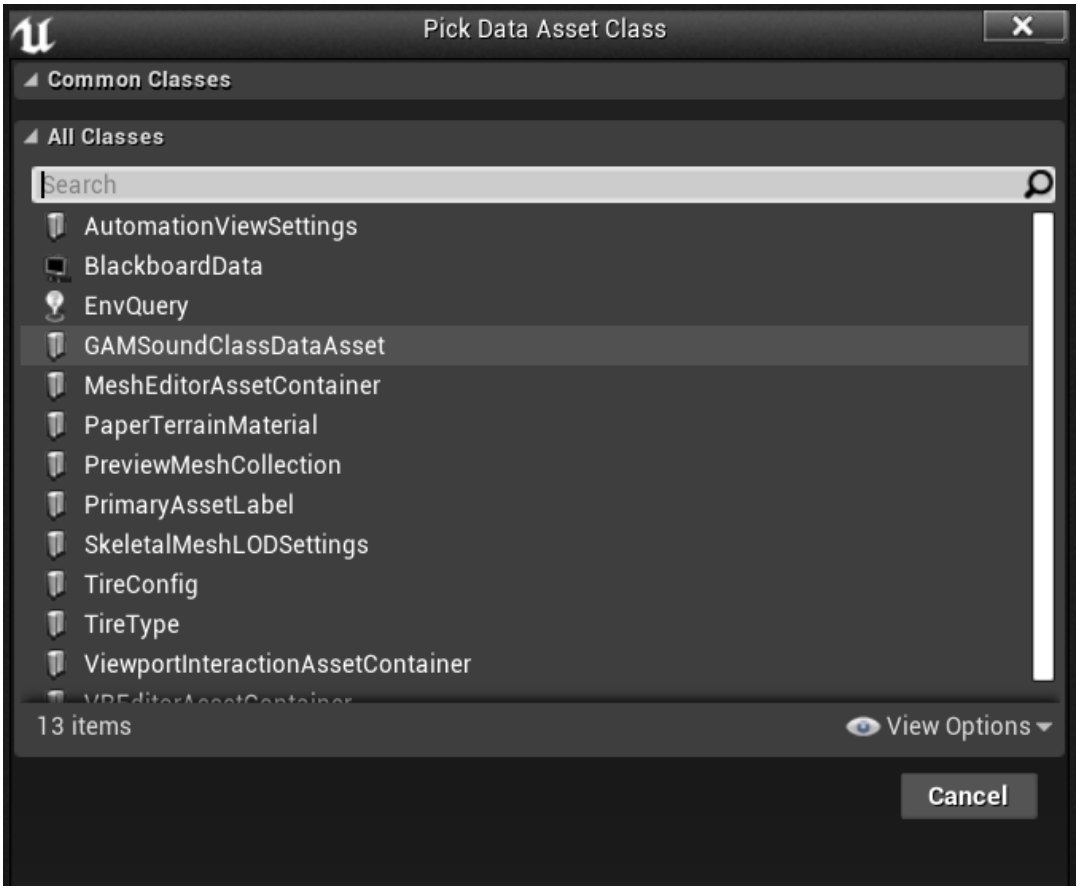
4. Setting up the Sound Class Audio Ducking

First of all, create a Player Controller which inherits from GAMPlayerController and add it to your Game Mode:



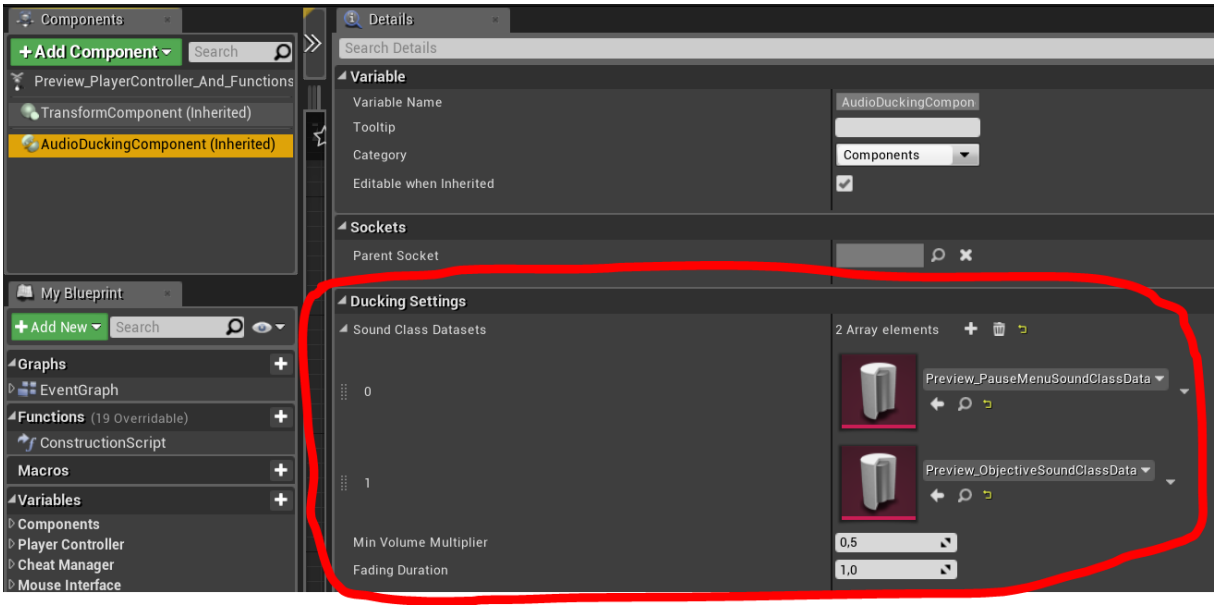
Now, create a Sound Class Data Asset which stores Sound Class assets:


Save Current
Source Control
Content
Marketplace
Perspective
Lit
Folder
New Folder
C++ Class
New C++ Class...
Import Asset
Import to /Game...
Create Basic Asset
Blueprint Class
Level
Material
Particle System
Substance
Create Advanced Asset
Animation
Artificial Intelligence
Blendables
Blueprints
Materials & Textures
Media
Miscellaneous
Paper2D
Physics
Sounds
User Interface
Drop files h
Composite Data Table
Curve
Curve Atlas
Data Asset
Data Table
Foliage Type
Force Feedback Attenuation
Force Feedback Effect
Haptic Feedback Effect Buffer
Haptic Feedback Effect Curve
Haptic Feedback Effect Sound Wave
Landscape Grass Type
Matinee Data
Object Library
Preview Mesh Collection
Slate Vector Art Data
String Table
Compile
Play
10°
0,25
4
Level: Untitled (Persisten
Content
Search Content
und Attenuation
Sound Clas



Each Sound Class has to have its own tag:

Now, add the Sound Class Data Asset to Audio Ducking Component in your Player Controller. You can add several Data Assets if you want:



PARAMETERS OVERVIEW:

SoundClassDatasets – Data Assets which store Sound Classes to fade in/out.

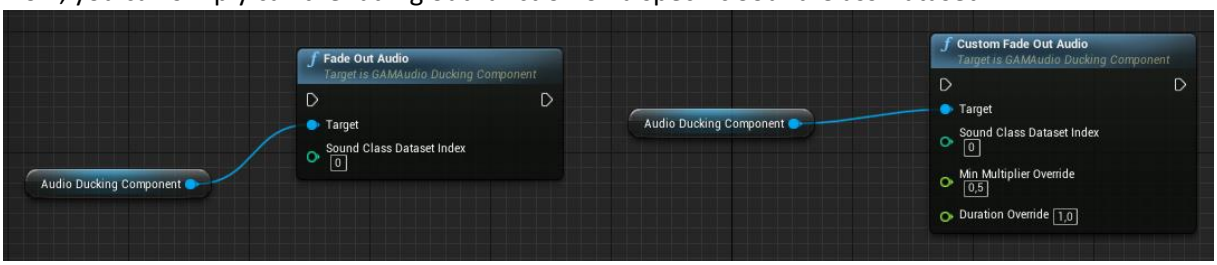
MinVolumeMultiplier – Describes how much Sound Classes should be faded out.

FadeVolumeLevel = DefaultVolumeLevel * MinVolumeMultiplier.

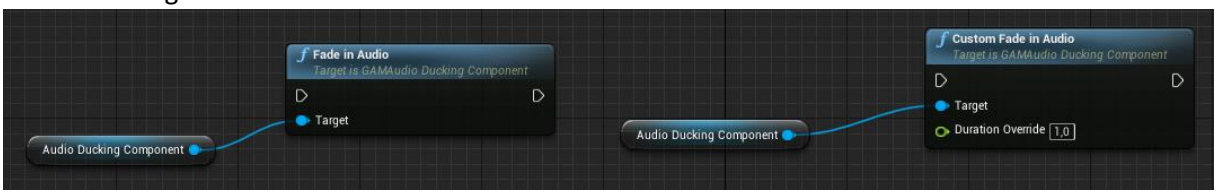
FadingDuration – The duration of fading.

5. How to use the Sound Class Audio Ducking

Now, you can simply call the fading out function on a specific Sound Class Dataset:

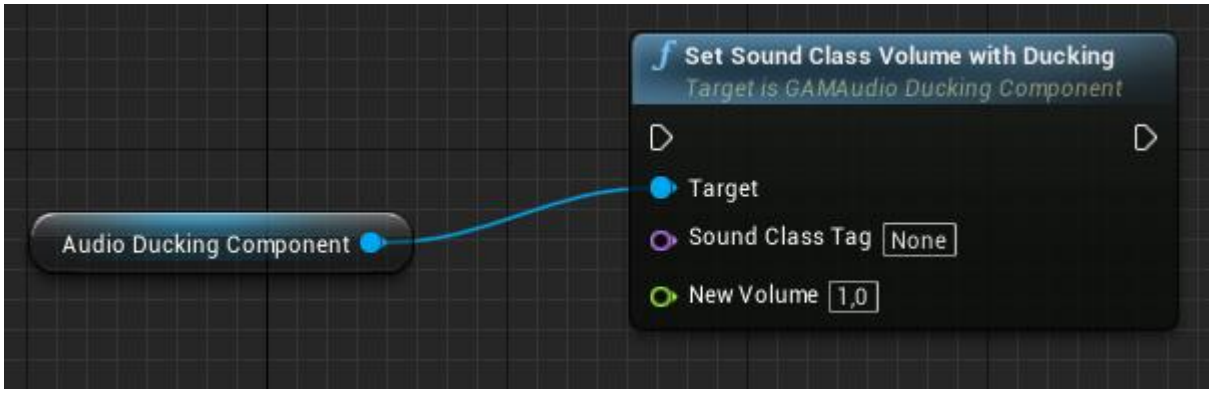


And the fading in function:

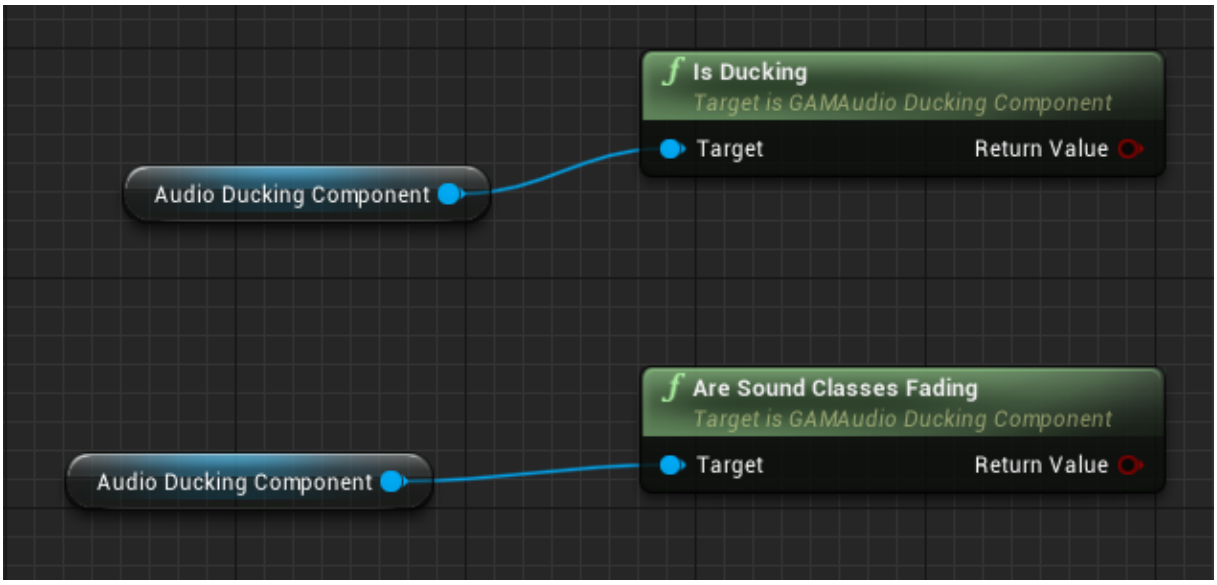


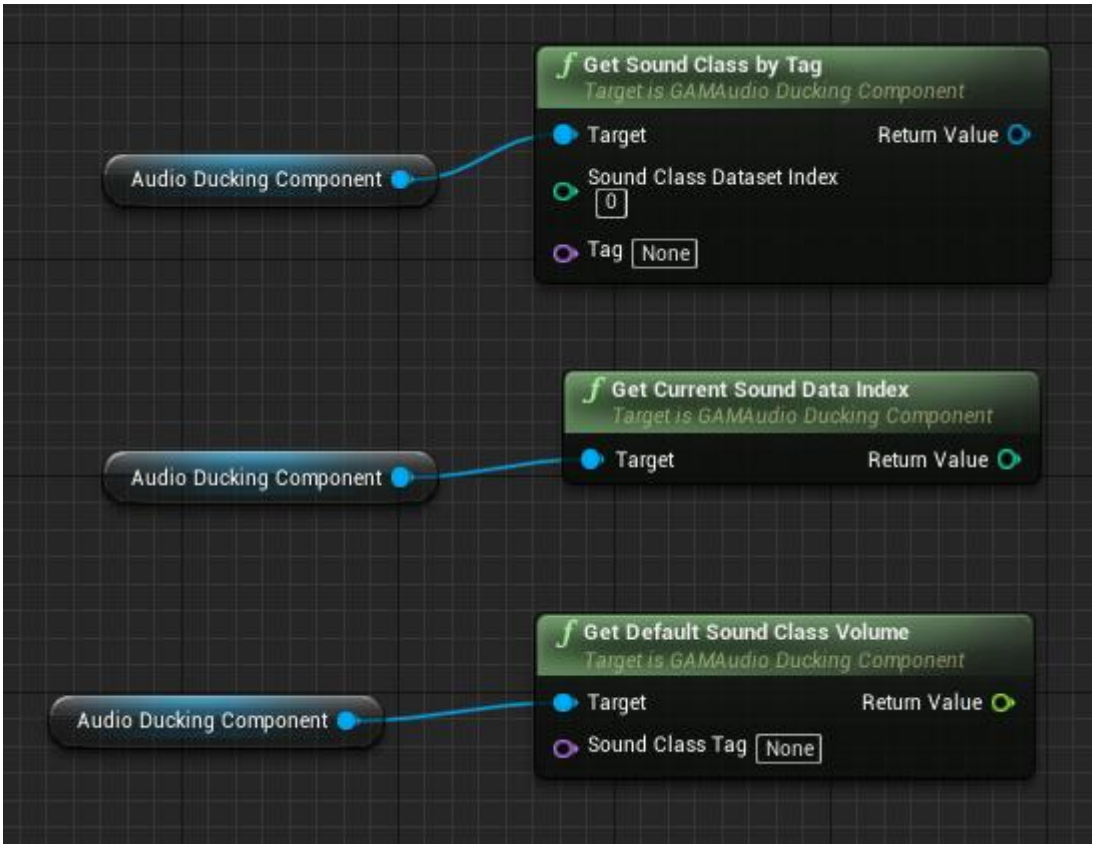
"Fade Out Audio" and "Fade In Audio" will use properties associated with the given Audio Ducking Component. "Custom Fade Out Audio" and "Custom Fade In Audio" will use overridden properties.

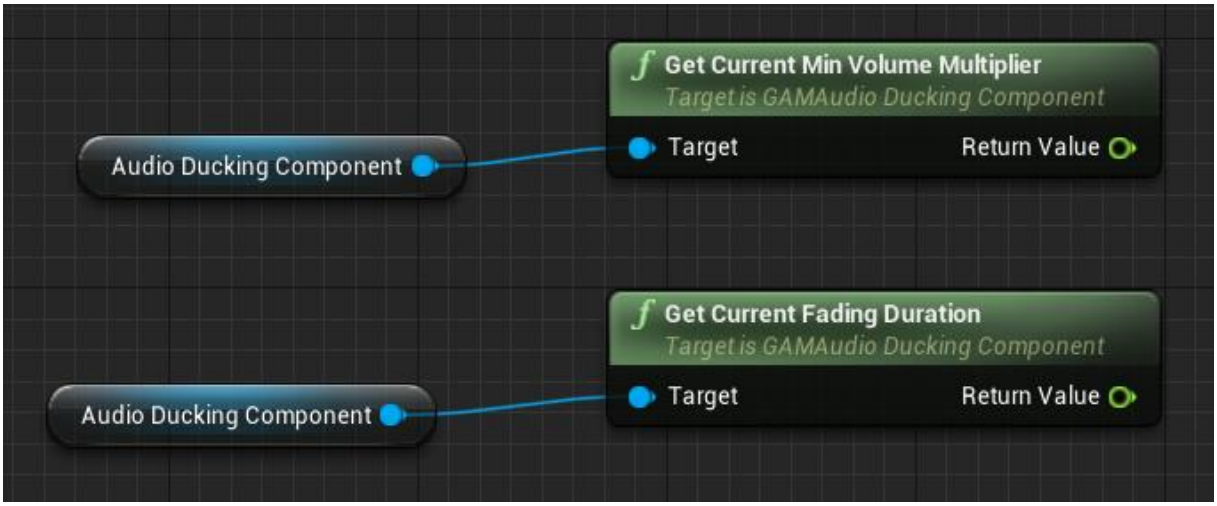
If a Sound Class is currently faded out and you want to change its volume, use "Set Sound Class Volume With Ducking":



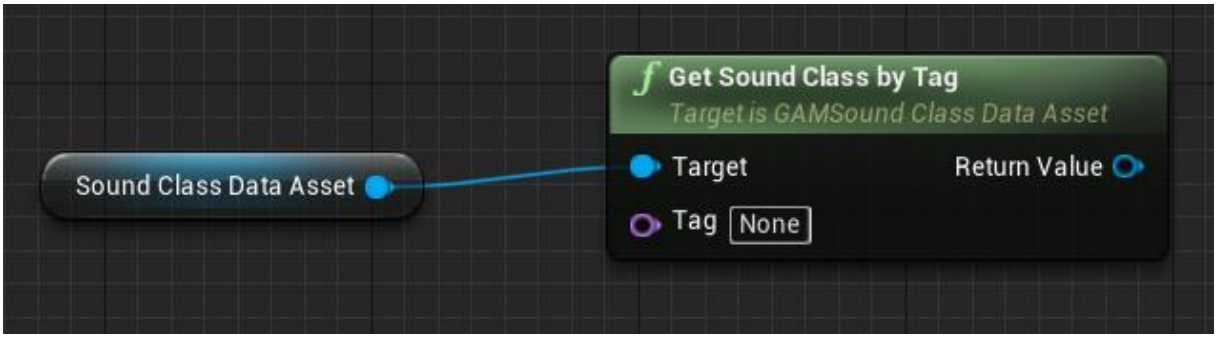
There are also some getters to receive ducking properties:





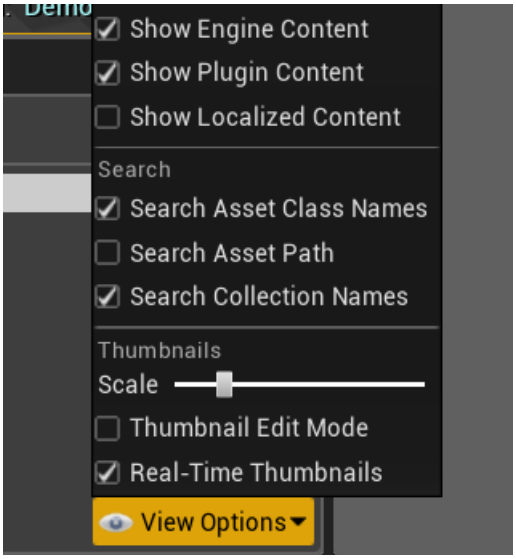


You can also get a specific Sound Class from the Sound Class Data Asset:



6. Preview content

To see a preview of how those functionalities work, open the **GlobalAudioManager\Content\Preview** directory. To do so, make sure you enabled "Show Engine Content" and "Show Plugin Content" in View Options:



- **Preview_PlayerController_And_FunctionsPreview** blueprint has a preview of all GAM functions
- **Preview_GameInstance** has a preview of initializing properties of Sound Classes
- **Preview_ObjectiveWidget** shows how to use fade in/out functions
- **Preview_OptionsWidget** shows how to build an Option widget by using GAM
- **Preview_PropertiesPreviewWidget** prints Sound Class and ducking properties
- There are also some preview Data Assets in **GlobalAudioManager\Content\Preview\DataAssets** directory

If you want to test all functionalities from GAM, open and play **GAM_DemoLevel** level.

7. Android build configuration

If you want to properly change pitch or reverb properties in an Android build, you need to change the **AndroidEngine.ini** file from **Config** folder. You have to use AudioMixerAndroid, instead of AndroidAudio:

```
[Audio]
; AudioDeviceModuleName=AndroidAudio
; Uncomment below (and comment above) to use multi-platform mixer module by default
AudioDeviceModuleName=AudioMixerAndroid
```

More information about the new Audio Engine can be found here.