

UniMod Guide



There are two scripts that serves as alternative to the “FMOD Studio Emitter”. Those are “Play FMOD Sound” and “Play FMOD MultipleSound” Both these scripts contains commonl FMOD sound triggering methods and adds to this a “PlayOneShotWithParameter”. These scripts differ from the “Studio Emitter” in that they don’t include “OnTriggerEnter” options and similar. While these functions in the FMOD script can be useful, the intention of UniMod is to aim for a modular approach, which is a more common code practise in Unity.

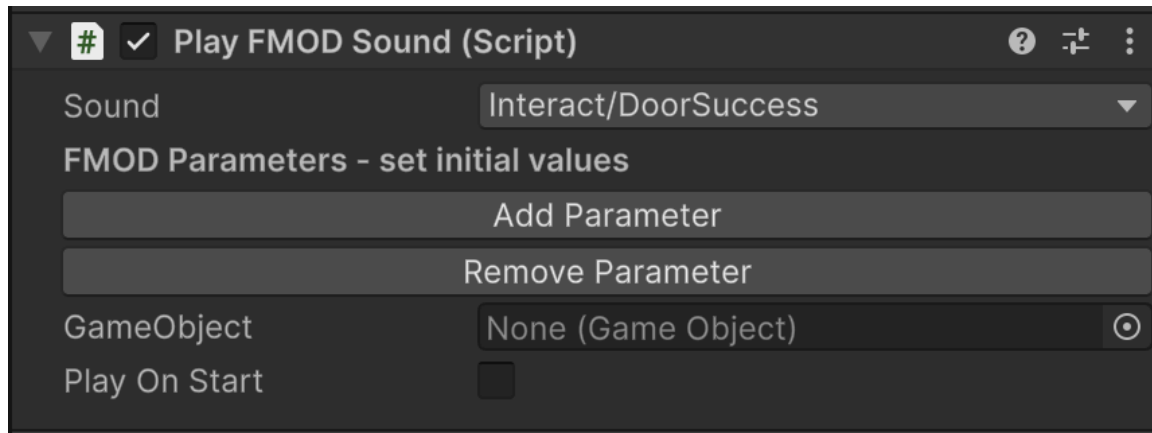
The methods available in both scripts are:

Method	Outcome	Comment
PlaySound()	Play instance	3D, Start and stop, parameters
PlaySoundAttached()	Play instance	3D, sound will follow game object trajectory
PlayOneShot()	Play and forget	3D
PlayOneShotWithParam	Play and forget	3D, params
StopSound()	Stop and release	N/A
SetParameter(string parameter, float value)	For FMOD multiple add the sound int (number)	N/A

Note that future updates may include a debug option such as parameter out from FMOD.

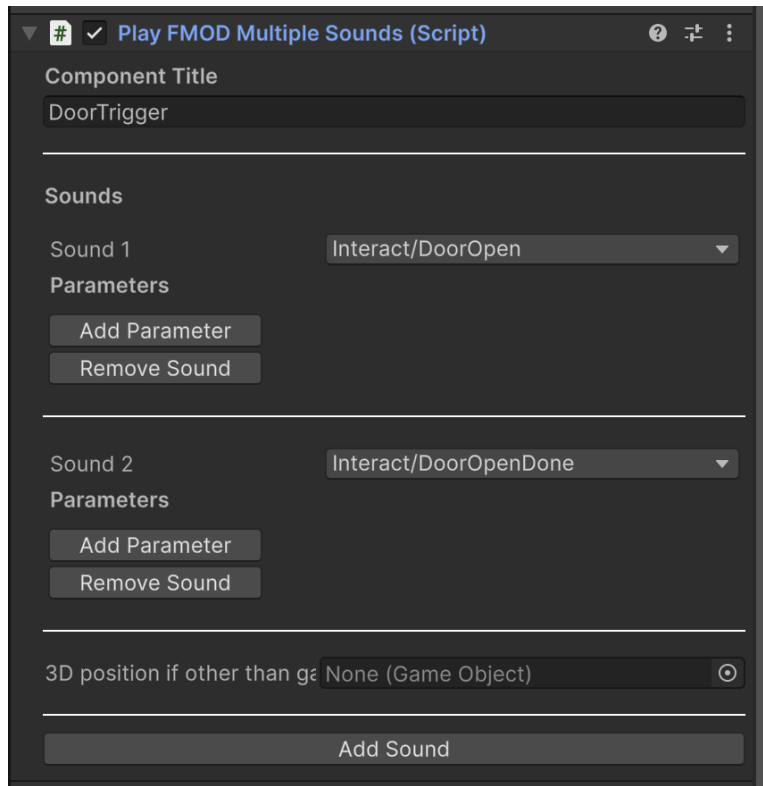
How to use:

Play FMOD Sound



- Create a scriptable object (So) found under Create/Audio/Fmod Sound. It's suggested to start by creating a "Dummy" So which can be used for programming purposes to avoid null errors that can happen if no sound is attached to the script/s.
- Add categories, 3D specification and any applicable comments.
- Add the FMOD Sound Manager script to an empty game object in the scene and populate the array "Sounds" with your So's.
- There's no need to categorise the sounds as this is only necessary when exporting via Excel.
- Add a "Play FMOD Sound" to your logic e.g., OnTriggerEnter, or simply set the sound to start via the checkbox on the script. Select a sound from the drop-down menu and you're done.
- GameObject - set a 3D position different from the transform the script is attached to.

Play FMOD MultipleSounds



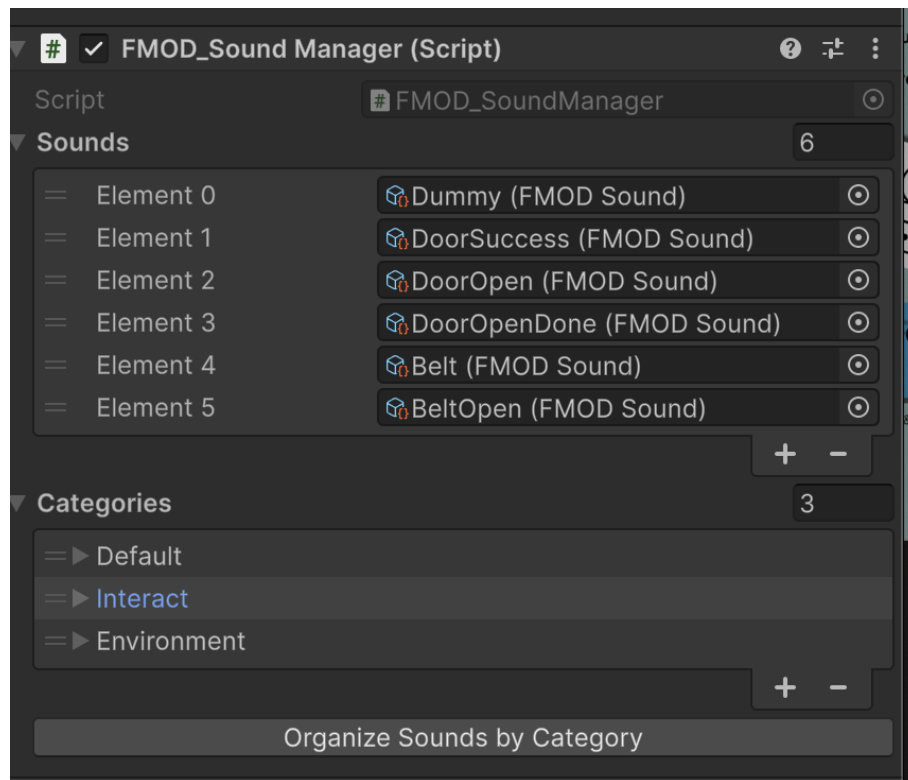
A common example on when to use this script is for a Player character footsteps, jump, land sound, etc. The different sounds on the script are distinguished via a numbering system (int) that starts at "0". In code, this is:

```
var Play FMOD Sound reference e.g., playSound  
playSound.PlayOneShot(0)
```

For additional sounds just increase the int i.e., `playSound.PlayOneShot(1)`, etc.

For parameters, the code is: `playSound.SetParameter(int, string, value)`.

FMOD Sound Manager



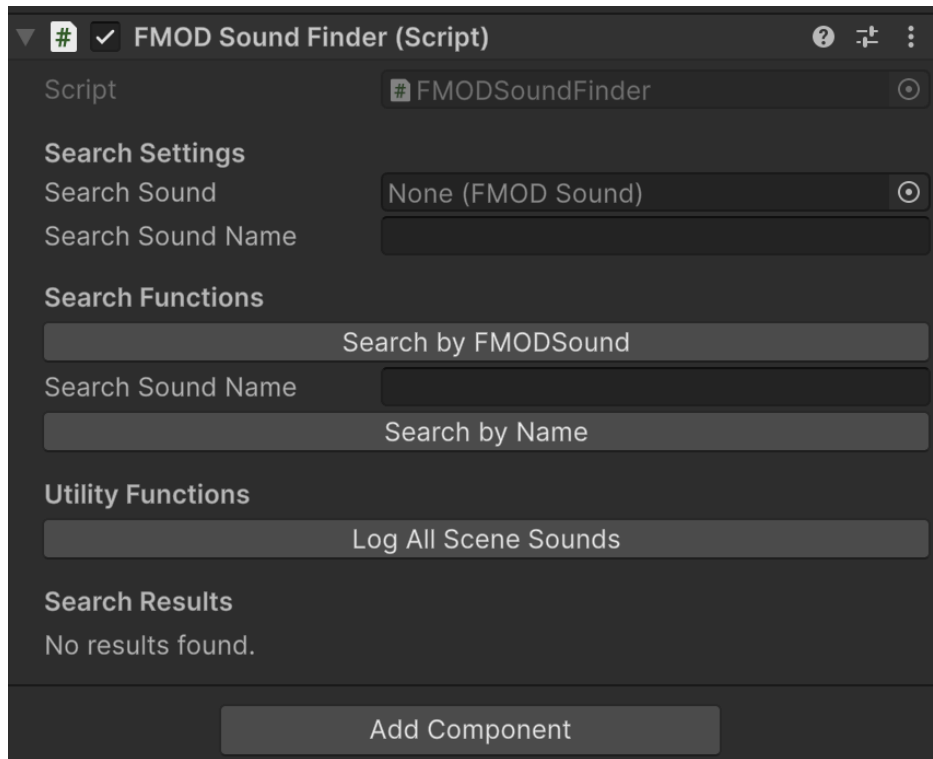
This script stores and keeps access to the So's. The different scriptable objects are categorised by default, and the "Organise Sounds..." button allows this to be visible in the Inspector also. This can be useful to allow collaborators to get an overview of what sound/music have been added to the development that requires creation and design.

Export Sound List To CSV

This function is found under the menu option "Tools" -> Export Sounds To CSV. It's advisable that the sounds have been categorised on the FMOD Sound Manager before export.

Category	Sound Name	3D Sound	Event Path	Comment
Default	Dummy	Yes	event:/DoorSuccess	
Interact	DoorSuccess	No	event:/DoorSuccess	
Interact	DoorOpen	No	event:/DoorOpen	
Interact	DoorOpenDone	No	event:/DoorOpenDone	
Environment	Belt	No	event:/BeltMoving	
Environment	BeltOpen	No	event:/BeltOpen	

FMOD Sound Finder



This script is found in the UniMod scripts folder and allows for finding where a sound is attached to a game object. Simply drag a So into the relevant box and press the “Find FMOD Sound” box. The corresponding game object will be shown at the bottom of the script in the inspector.

You can also use “Log All..” to attempt to get all references.

Note that depending on where the scripts are being called/located in your scene, you might get odd/no results. This is because it's difficult to predict exactly how a scene might look and also because this script is meant as a broad helper particularly for the sound/music designer and not as a specialised tool for precise analysis.