Bug Report [SOLVED] – 08/31/2021 – C++ Core Audio Engine and C# integration

* Some components have elements that prevent from building the application (on gizmo draw in all players, the whole of speaker config script) - temporary fix by using "if unity editor" for editor only scripts.

[FIXED]

*Comment : I used the same method with pre-processor directive : #if UNITY\_EDITOR*

* The file browser prevented from building which was fixed by changing the project to .net 4.xx

[FIXED]

*Comment : (thanks !)*

* state and external assets should be made so that the build and editor have independent paths - same for sound card selection. This would be helpful for the creation process and then the file structure can be duplicated in the final build location with state files.

[FIXED]

*Comment : you can now add two other folder references for states and audio, whose are used for standalone execution. However, changes are limited in the json file. For the output (first line of the state), you are just allowed to change the sampling rate. For the players you are just allowed to change the relative path of the file name of "type":0 (At\_Player class)*

* player keeps "playing" when audio file ends, when play is called again it is not possible since it seems as if its playing. Need to manually stop the player and restart. Could be managed in both player and random player so that random instancing can work properly.

[FIXED]

*Comment : you can used the public method StartPlaying(), StopPlaying() and SafeDestroy(), to start, stop and destroy the At\_Player. For the random player use the method AddOneShotInstanceAndRandomPlay(). /!\* At\_DynamicRandomPlayer *class now manage instances differently. Unity3D has strong problems with Garbage Collection. Memory is not totally released when destroying an object, which cause crashes of the game engine when instantiating during a long time at a strong rate. Now there is a fixed size pool of At\_Player instances, providing slots used to play a new audio file chosen randomly. The size of the pool is set to 20 instances. You can change the initialization of the constant variable “*maxInstance*“ if you want to increase or decrease it. Fill free to adjust it depending on performances (memory, CPU, etc.). This size is also the same for each* At\_DynamicRandomPlayer *instance. It could become a parameter expose in the editor for more fine settings if you want.*

* Random Player loses reference to all sounds during runtime and can not play anything but when you click back on the object it finds the reference and loads the sounds. when this happens it keeps the player settings but not the files - and still nothing is playing when the references are found

[FIXED]

*Comment : I spent time to clean references of players with guid. This should be all clean now. You can destroy, remove component, duplicate GameObject. The json file should be always up to date and I normally avoid unexpected object removals.*

* there is a strange clicking in the speakers when the audio is moving from the zone of one virtual mic to another. (when audio is changing channels)

[FIXED]

*Comment : I add overlap-and-add for delay and gain in the core spatialization engine to avoid “clicks” when positions of object are changing to fast. This should be clean now.*