Elias Software

Elias 3 plugin for Unity

Plugin version 3.2.2

www.eliassoftware.com

Table of contents

1   Important Information

If you use Elias in a commercial product you will also need to buy a licence for the Elias Engine. For more information on engine licences and prices visit https://www.eliassoftware.com/licensing/.

2 Install the Elias 3 Plugin

If you have an Elias 1.x plugin already installed to your Unity project, be sure to remove it before installing the Elias 3 plugin.

Install the Elias Plugin for your Unity Project by running the "EliasUnityPlugin.unitypackage" while your project is open. This will add three folders to the Assets folder associated with your opened project.

The Elias Demo folder contains a demo scene that you can open up and see an example of Elias in action. This folder can safely be removed.

The Plugins folder contains the Elias Unity Plugin scripts and engine binaries.

The "StreamingAssets" folder is a place for you to place your Elias Themes and their files. When you don't need the demo scene or demo music you can remove them from your project. Note that the "StreamingAssets" folder is handled differently by Unity, and anything inside it will always be deployed as is.

3   Get started with Elias in Unity

3.1   Step by step

1. When you are ready to import your Elias project to Unity you can first start with exporting a delivery package from the Elias Studio. If you are working with Wav files, it can be a good idea to convert them to Ogg Vorbis. In the export delivery package dialogue you can set the approximate bitrate of the Ogg files. You can also get help organize your project files by using automatic naming of your files and compress them into a zip archive. The mepro will not be modified unless you change the format or choose to rename the files, in which case the paths will be updated.

2. Add your Elias project to the Unity project folder and inside Assets/StreamingAssets/Elias_Themes. Notice that the project may not be in a zip file.

3. Make sure that the Elias plugin has been installed to your Unity project and add a game object to the scene. Add the Elias Player component to the new game object.

Let us go through the Elias Player from top to bottom:

4. Script - This is the "Elias Player" script

5. Elias Channel Count - Set the number of channels you want Elias to output. Select "1" for mono, "2" for Stereo and "6" for 5.1 etc. If Elias uses 4 or more channels (quad), please set channel count to 0 allowing the Elias Engine to match Unity's channel count internally.

6. Play On Start - Check this box to start playing Elias when the level starts.

7. Read more about frames per buffer, cache page count and bytes per cache page here: 5 Cache and buffer

8. Sample Rate - Set what sample rate your audio files use and notice that this is set separately from Unity's sample rate settings.

9. Project - Select what Elias project (mepro file) you want to use.

10. Action Preset - If you want Elias to start playing with an action preset, select it here.

11. Wait Time (ms) - To make Elias quiet for a set time from the start, set a wait time in milliseconds here.

12. Transition Preset - Select which transition preset.

13. Theme - Select which theme you want to start with.

14. Jump To Bar - If you want Elias to start at a specific bar in the loop, set it here.

15. Track Group - If you want to start with a specific Track Group, select it here.

16. Level - Select which level you want to start playing.

17. Suggested Max Time - Set how much time Elias should listen to determine the best transition point.

18. You can now remove the Elias Project that comes with the plugin called Plugin Demo Theme if you do not need it anymore, since everything in your "StreamingAssets" folder will be copied into the game.
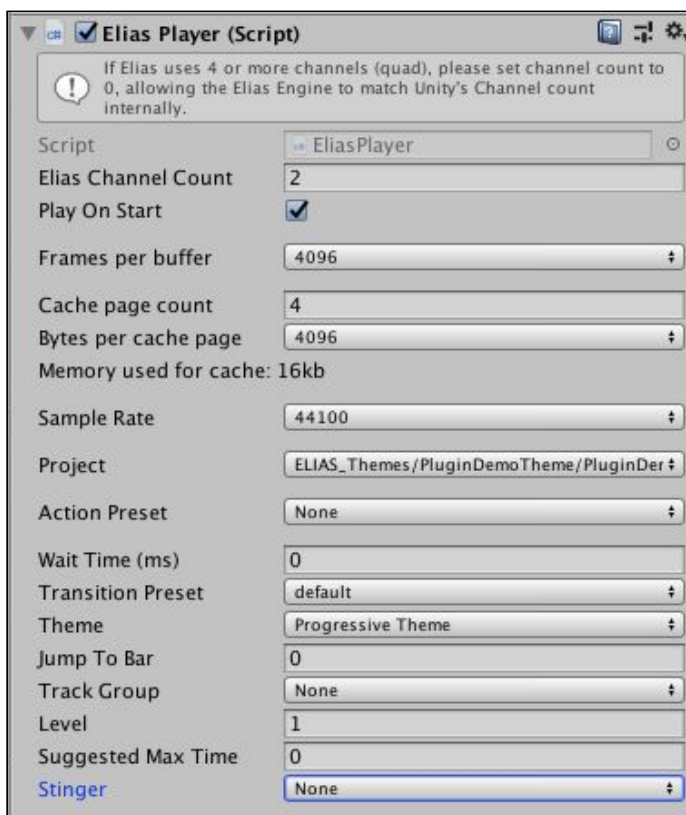
The Elias Player component that comes with the plugin.

4   Urgent and wait for prior events flags

4.1   Urgent

Using the Urgent flag will bypass the event queue to perform the action as fast as possible. For example, if you know that your project will have a lot of level changes and be very busy, the event you want to run can be delayed because of the overall queue of events in Elias. In those cases, be sure to check use urgent to prioritize this event in the queue.

4.2   Wait for Prior Events

If the flag "Wait for Prior Events" is not used, this event can be skipped to reach the final destination of the Action Preset as fast as possible. If you want to make longer series of events and use multiple set level events, be sure to check the "Wait for Prior Event" flag to make sure that the event will run before moving on to the next event in the chain.

5   Cache and buffer

If you have performance issues, you can adjust "Frames per buffer", "Cache page count" and "Bytes per cache page" and find the right settings for your project. Below the controls you can see how much memory is used for the cache with the current settings.

Using larger buffer chunks per frame will have a smaller impact on performance, but will instead introduce more latency, so depending on what platforms you are building for and how heavy your Elias project is to run, these settings can be varied.

Frames per buffer	4096
Cache page count	4
Bytes per cache page	4096
Memory used for cache: 16kb	

*Cache and buffer settings as seen in the Elias Player component.*

6   Demo Scene

In your assets folder, locate the Elias Demo folder that contains a demo scene with examples. If it sounds like the music is running too fast or too slow, make sure you have set your System Sample Rate to 44100 in the Edit menu under Project Settings and then Audio so they match the audio files in this project.

Moving between the rooms will switch themes, and the switch occurs quickly when going through the triangle doorway, and more slowly yet more musically when going through the doorway with an arch. Going through the arches triggers two different Action Presets that use two different Transition Presets that makes the transitions different from each other.
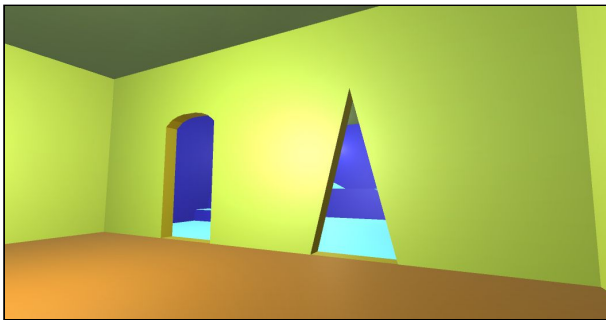
The slopes have three green parts, the floor level will bring the theme to Level 3, the middle one to Level 7 and the top one to Level 10. The level changes are triggered by colliders placed in these areas with the Elias Demo Event Trigger Script.
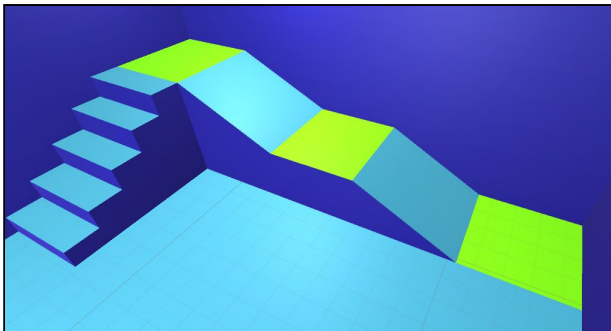
The blue room also has two spheres that will either play a tonal stinger or a Game Over Action Preset upon colliding with them. The Game Over Action Preset will bring the theme to level 0 making it go silent. The theme will start playing again if you walk onto any of the green parts or change theme by going through one of the doorways. The spheres uses the Elias Demo Event Trigger script to run the Action Presets

Feel free to make changes to this scene and experiment with the settings to learn more about using Elias with Unity.

6.1   Elias Demo Event Trigger

WARNING: This component is a part of the Demo Code and not guaranteed to be compatible between versions of the plugin! It is meant as an example on how to queue/use different types of events.

Add this component to an object or event to give instructions to the Elias Project on what you want to change. Here you can set level, play stingers, use Action Presets, set effect parameters and set volumes. Check the associated check box and allow the Event Trigger to make the changes you want it to make.

If you want to use a Stinger in a Set Level event and if you at the same time change theme you specify the theme that the stinger belongs to at the Stinger Required Theme row. If you know that the theme that this stinger belongs to is already playing you can leave that row empty.
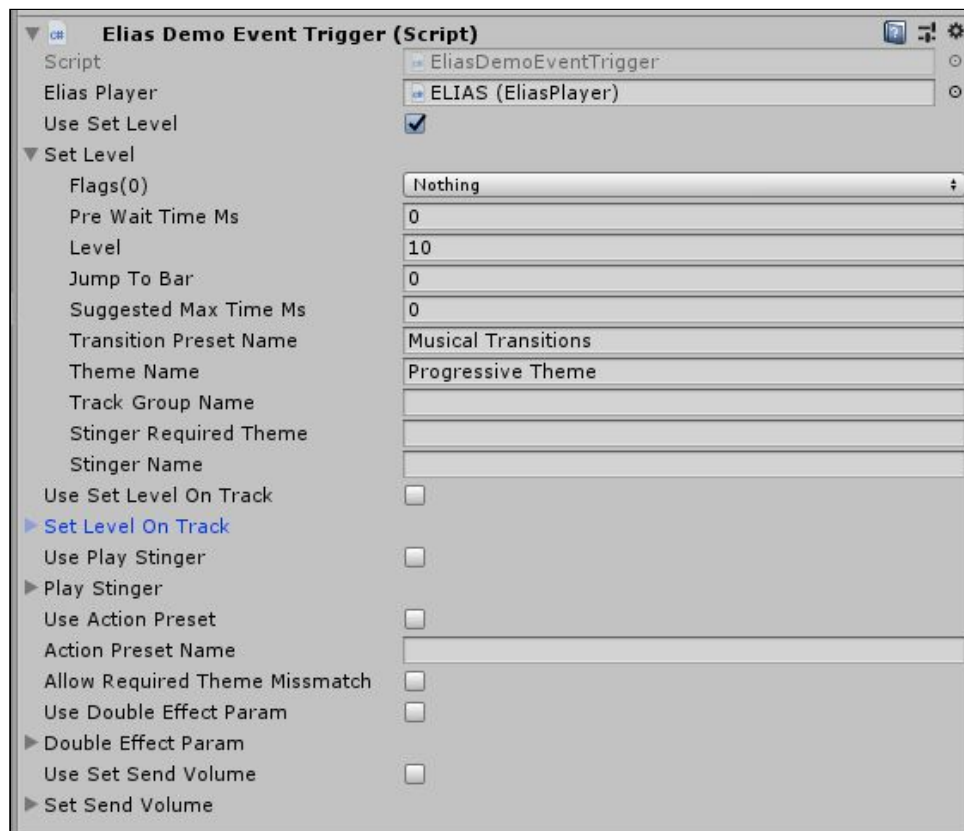
In this example screenshot we can see that this trigger will make the theme go to level 10, using the "Musical Transitions" Transition Preset and will do this in the "Progressive Theme" in the Plugin Demo Theme Project.

7   Tips and tricks

- To save you a lot of work and set up in Unity, use Action Presets as much as you can. Action Presets can easily be set up, tested and adjusted inside the Elias Studio and a lot of steps and instructions are made in just one simple call from Unity. This also makes it easier for the composer/sound designer to modify the events without having to modify the scenes.

- The Elias Player and the Elias Demo Event Trigger components can take you far, but sometimes you might need to give Elias instructions from a script. You can find help in the EliasHelper, EliasSetLevel, EliasSetLevelOnTrack, EliasSetEffect and EliasSetSendVolume scripts. Find the scripts at: Assets/Plugins/Elias/Scripts.

- If you are using midi tracks and patches in your theme and it plays as it should inside Unity but not in a build for iOS or Android, the problem is most likely that the paths to the samples in the sfz file needs to match the directory and file names exactly with the upper and lower case letters.

8   Change Log

8.1 Updates in version 3.2.2

- Elias sample rate is now set separately from Unity's own sample rate settings.
- Set channel count for Elias
- In the event that the number of output channels in Unity doesn't match the number of output channels in Elias, Elias will now handle such occurrences.
- Improved get Action Preset Events.
- General improvements and bug fixes.
- More engine code has been exposed to the plugin for more control and accessibility.

8.2 Updates in version 3.2.1

- Added Arm64 for Android for Unity's 2017.4 LTS release.

8.3 Updates in version 3.2

- Added stingerRequiredTheme in setLevel och setLevelOnTrack events.
- Made some significant performance optimizations in the SFZ player and the general mixer rendering chain in the engine.

8.4 Updates in version 3.1.1

- Added a new flag that can be set for events, called elias_event_flag_wait_for_prior. When this flag is set, the engine ensures that all prior events have finished processing before it proceeds.
- Improved performance in the Elias Sampler for midi playback.

8.5 Updates in version 3.1.0

- Significantly optimized the performance of the audio output rendering.

8.6 Updates in version 3.0.2

- Added threaded rendering to the Elias engine, allowing us to double buffer the Elias output. This results in much improved performance tolerance. Overall this will result in the latency being increased, in exchange the risk of stuttering is significantly lower, even with smaller buffer sizes.
- Added more tools to handle the cache: "Cache page count" and "Bytes per cache page".

8.7 Updates in version 3.0.1

- "Fixed a race condition if the AudioSource was started before Elias was.
- Updated mepro file in demo scene.

8.8 Updates from version 2.1.1 to 3.0.0

- Added Set level on Track Event
- General bug fixes
- Improvements to handle midi files and instrument patches.
- improved the engine for handling midi, sampler and instrument patches.
- Fixed a rare crash when rendering failed due to corrupted ogg files.

8.9 Elias Engine 3.0.0 fixes and features

- Fixed a bug in the MIDI file streaming implementation which resulted in some old notes continuing to play after transitioning away from the given level or variation.
- Fixed a bug where the engine would occasionally fail to clean up SFZ voices properly.
- Many internal optimizations in the SFZ player.
- Fixed a bug in the third party Ogg Vorbis decoder that caused certain seeking operations to fail.
- Added a temporary workaround for a bug in the third party Ogg Vorbis decoder which causes it to skip the last Vorbis frame in certain files. The engine currently mutes the remainder of the buffer when this problem occurs; a fix is in the works by the author of the decoder.
- Fixed a bug which resulted in lost MIDI events when sending data from multiple MIDI tracks to the same generator.
- Implemented so called "smart transitions" based on pre analysis of the audio files. When these are enabled, the engine will generate significantly smoother transitions between levels and variations by taking advantage of very quiet portions of the audio being rendered. Smart transitions can be disabled even if analysis data is available, by way of the elias_transition_option_disable_smart_transitions boolean transition option.
- The engine can now be instructed to automatically transition to the next variation on a level based on the given progression mode each time the loop wraps around. This is configurable by way of a new boolean transition option called elias_transition_option_enable_automatic_variation_transitions. This works both for audio and MIDI tracks, but only if the level in question contains no silent variations.
- Implemented a new "level selection strategy" called elias_level_selection_exact.