Midi Player Tool Kit Documentation Version 1.1

Midi Player Tool Kit is a package for Unity able to play music from Midi files in your application. There is no need of external component or device.

Thank to SoundFont for adding hundred sounds of instruments for playing Midi files.

- No scripting C# or JS is mandatory: specific inspector defined all your needs directly in Unity editor. See Quick Start.
- Add Midi Audio Source Prefab in your scene as much as you want.
- Add Midi files (from the web or from your composition) as much as you want.
- Play / Paused automatically based on distance.
- API available to integrate MPTK in your application.

Midi

Midi file is a well-known format for playing music. It's very easy to find a lot of varied music often for free on the web.

The benefits of Midi is: smaller compared as classic audio files, more flexible (transpose, speed, ...)

Try here to find interesting Midi file :

- http://www.midiworld.com Midi World is a huge free Midi site with varied genre : classic, pop, rock, rap, dance, punk, blues, country, movie themes, tv themes, christmas carols, video game themes, disney themes, national anthems, jazz , hip-hop.
- https://freemidi.org Free Midi is also huge free midi site with varied genre : Rock, Pop, Rap, Hip-Hop, R&B, Soul, Classical, Country Folk, Jazz, Blues, Dance / Electric, Folk, Punk, Newage.
- https://www.vgmusic.com Huge library of Midi video game. Started in 1996, thousands Midi files are available.
- http://www.midishrine.com Another site specialized in Midi video games.

Tips :

- Don't forget to ask authorization for copyrighted resources!
- If you or a friend, are musician you could also compose your own music, record as Midi and use it in your application!

SoundFont

Midi Player Tool Kit is based on SoundFont, a kind of library which contains hundred sounds of instruments as piano, sax, drum, ...

But SoundFont are designed to be used with synthesizer or powerful computer. Midi Player Tool Kit converts SoundFont to a simpler format with not all the functionalities of SoudFont but enough to play music to support application in Unity.

The FREE version of Midi Player Tool Kit use the SoundFont GeneralUser_GS_SoftSynth_v144. It's a sound font with a good balancing between size and quality.

Thousands SoundFonts are available on the web, often for free, the PRO version is able to use another SoundFont and optimize the size of sounds in your application.

Find SounfFonts here:

- http://trisamples.com/free-soundfonts/
- https://cymatics.fm/blog/soundfonts/

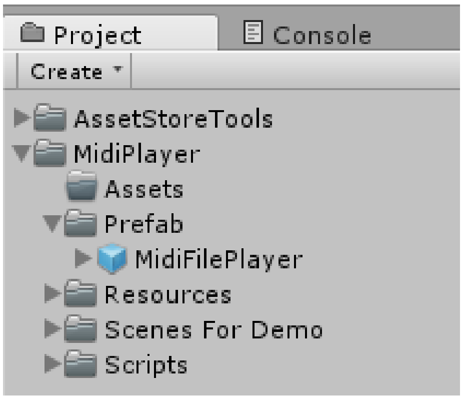
Tips : try General Midi SoundFont to get a large variety of instruments and a compatibility with Midi norm.

Quick Start

1- Project

1.1 - Download Free Midi Player Tool Kit or download Pro Midi Player Tool Kit (coming soon !).

1.2 - Import Unity package in your Unity application, you will get something like this in your project:

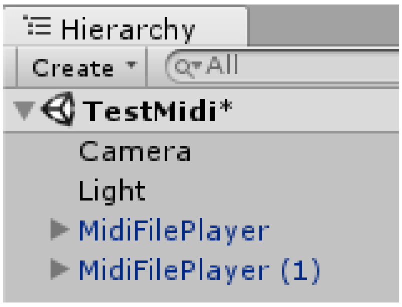


Tips : Free version comes with a default SoundFont (GeneralUser_GS_SoftSynth_v144) and some examples of midi files. Of course you can delete all Midi files and add your favorites (see Midi Player Tool Kit Setup below).

2 - Hierarchy

2.1 - Add to the hierarchy one or more "MidiFilePlayer" Prefab from "Asset/MidiPlayer/Prefab". Each one can plays a different Midi file.

2.2 - Check that you have an AudioListener component somewhere in your hierarchy, typically associated with the "Main Camera".



3 - Midi File Player Inspector

3.1 - Select one MidiFilePlayer in the hierarchy.

3.2 - From the Midi File Player Inspector:

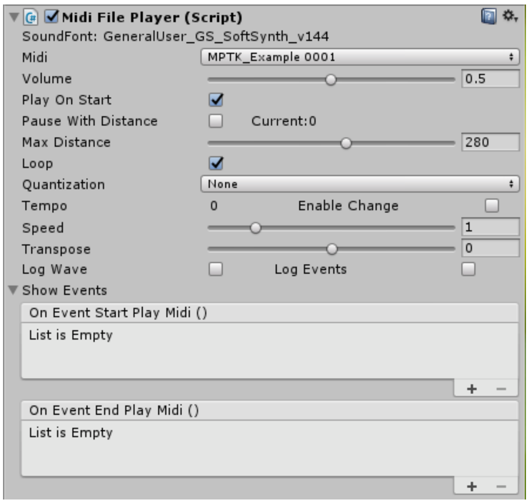
3.2.1 - Select the Midi File to play. The Midi Files available are defined with the tool "Midi Player Toolkit". See below Setup Midi Player. But some examples are provided with the free version.

3.2.2 - Check "Play On Start".

3.2.3 **It's over! Run your Unity application, the Midi begins to play!**

4 - More advanced used with the Midi File Player Inspector:

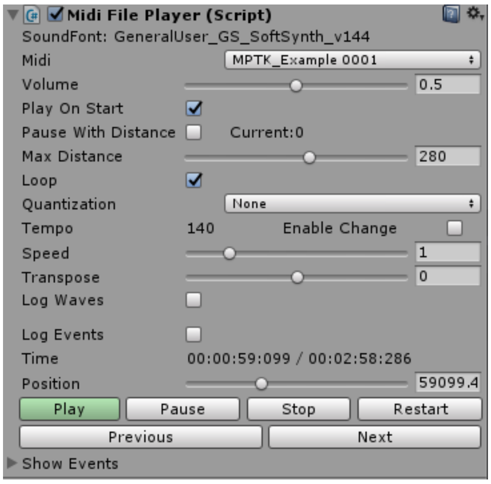
- "Play On Start" if checked, start playing when your application start.
- "Pause with distance" if checked, pause playing if distance between MidiFilePlayer GameObject and your AudioListener is greater than MaxDistance.
- MaxDistance. If "Pause with distance" is checked, Midi playing is paused depending the distance. See below Audio Source Template
- "Loop" if checked, restart playing automatically at end of the Midi file.
- Quantization: define quantization to remove some imprecision in tempo. With quantization, notes are plays on beats or on fractions of beats.
- "Enable Change" if checked, tempo change from the Midi are used (to be enabled if you encounters some issues when playing).
- "Transpose" transpose the music by half tone.
- "Events" set events to trigger actions in your scripts at start/end playing.



☑ Midi File Player (Script)
SoundFont: GeneralUser_GS_SoftSynth_v144
Midi: MPTK_Example 0001
Volume: 0.5
Play On Start: ☑
Pause With Distance: ☐ Current:0
Max Distance: 280
Loop: ☑
Quantization: None
Tempo: 0 Enable Change ☐
Speed: 1
Transpose: 0
Log Wave: ☐ Log Events ☐
Show Events
On Event Start Play Midi ()
List is Empty
On Event End Play Midi ()
List is Empty

Midi File Player Inspector in Run Mode

In Run Mode with the inspector, you can Play / Pause / Stop / Restart, change the current playing position, display midi events. It's useful to quickly check your Midi file.



☑ Midi File Player (Script)
SoundFont: GeneralUser_GS_SoftSynth_v144
Midi: MPTK_Example 0001
Volume: 0.5
Play On Start: ☑
Pause With Distance: ☐ Current:0
Max Distance: 280
Loop: ☑
Quantization: None
Tempo: 140 Enable Change ☐
Speed: 1
Transpose: 0
Log Waves: ☐
Log Events: ☐
Time: 00:00:59:099 / 00:02:58:286
Position: 59099.4

Advanced Use

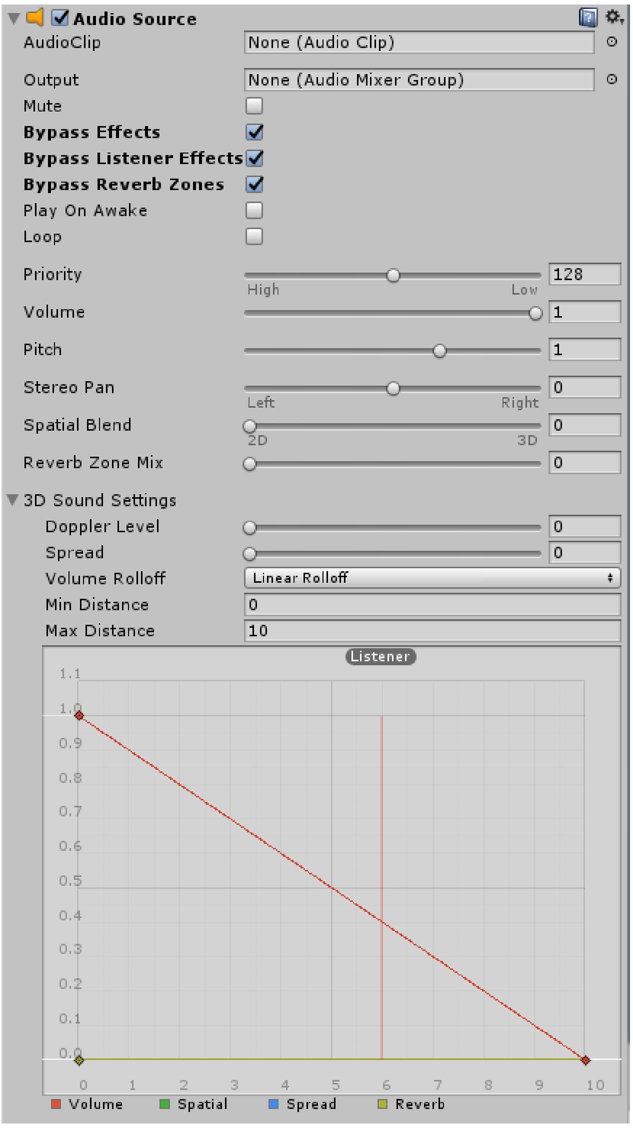
Audio Source Template

Each MidiFilePlayer gameobject contains an AudioSource template. This audio source is used as a template to play each sounds in your Midi File Player.

You can customize it to adapt to your need. Picture below is the default configuration deployed with MPTK. Go to Unity documentation to understand how to use it.

Tips:

- By default Spatial Blend is set to 0. Set to a upper value to change volume accordingly with the distance to the Audio Listener and the Left/Right orientation.
- Audio Source Template are created at run time when a new sound is need. So, change the template parameters at edit mode to be sure that changes will be taken into account for all sounds when playing.



☑ **Audio Source**

AudioClip: None (Audio Clip)

Output: None (Audio Mixer Group)

Mute □

Bypass Effects ☑

Bypass Listener Effects ☑

Bypass Reverb Zones ☑

Play On Awake □

Loop □

Priority: 128 (High – Low)

Volume: 1

Pitch: 1

Stereo Pan: 0 (Left – Right)

Spatial Blend: 0 (2D – 3D)

Reverb Zone Mix: 0

3D Sound Settings

Doppler Level: 0

Spread: 0

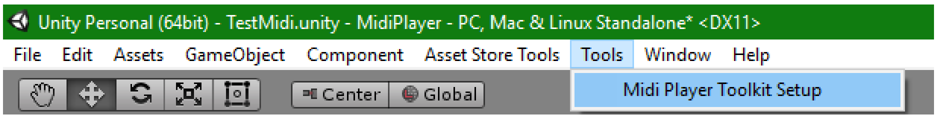
Volume Rolloff: Linear Rolloff

Min Distance: 0

Max Distance: 10

Set-up Sound Font and Midi File

Go to the menu Tools and "Midi Player Tool Kit" to display the setup


Unity Personal (64bit) - TestMidi.unity - MidiPlayer - PC, Mac & Linux Standalone* <DX11>
File
Edit
Assets
GameObject
Component
Asset Store Tools
Tools
Window
Help
Center
Global
Midi Player Toolkit Setup

There is 6 panels, 2 panels by rows, see screen capture below :

1. Top : manage SoundFont (Pro version)
2. Middle : manage your Midi Files (all versions)
3. Bottom : optimize SoundFont according your Midi Files (Pro version)

Panel SoundFont [Pro Version]

- "Add SoundFont": choose a SoundFont (filename of type .sf2) on your desktop and load it in Midi Player ToolKit. You can select the banks to used :
  - Default bank for sound as Piano, Sax, Organ, Guitar. Often the first bank contains a well selected variety of instruments.
  - Drum Kit bank for drum sound. Often the last bank is dedicated to Drum Kit.
- "Select": select the default SoundFont to use. Only one SoundFont can be selected at the same time.
- "Remove": delete the SoundFont and all sound associated to it.

Tips: Free Version is published with the SoundFont "GeneralUser_GS_SoftSynth_v144". It's a free sound font with a good balancing between size and quality.

- When publishing your application, removes unused SoundFonts to

Panel MidiFile [Free & Pro]

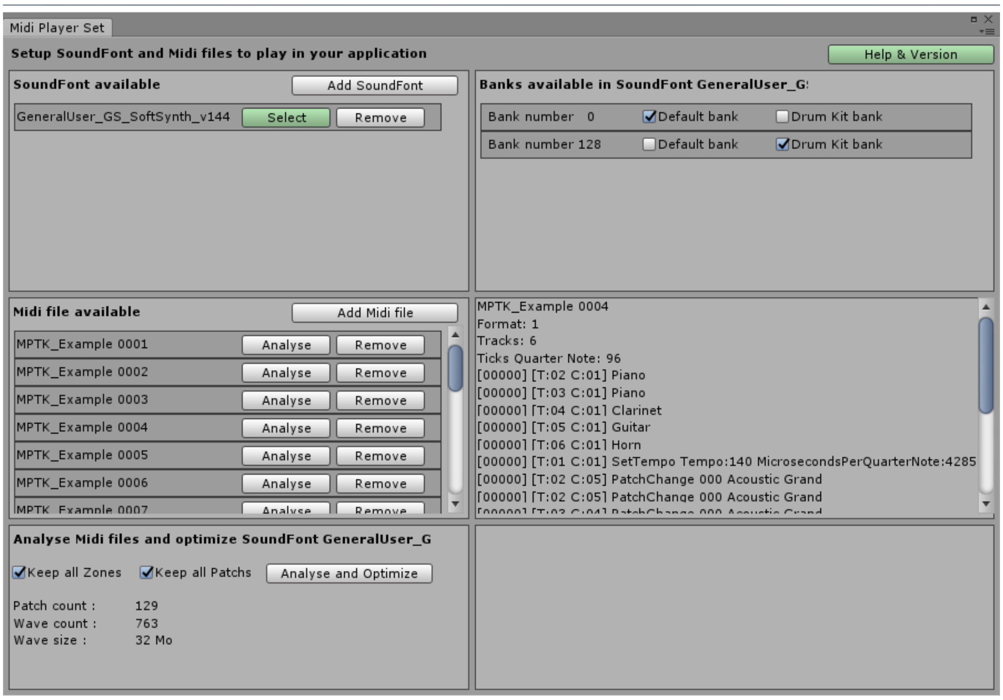
- "Add Midi File": Add a Midi File (filename of type .mid) from your computer to Midi Player Tool Kit
- "Analyse": Read the Midi File and displays informations in the right panel : tempo, patch, copyright, ...
- "Remove": delete the Midi File from the list

Tips :

- You can also directly imports a group of midi files in yout project : drag and drop Midi files into folders MidiPlayer\Assets\MidiPlayer\Resources\MidiDB. Next, you have to go to menu Tools and "Midi Player Tool Kit" to automatically add these new Midi files.
- Think to remove unused SoundFonts when publishing your application !

Panel Optimize [Pro Version]

- Click on button "Analyse Midi Files" : each Midi Files in the list are analyzed to detect patchs and waves used. Patchs and waves not used are stripped from the SoundFont to reduce the size of your project. A simpler format of SoundFont is created.
- "Keep all Zones" : if checked, all waves for selected patch are kept. It is useful if you want to transpose music when playing.
- "Keep all Patchs" if checked, all patchs are kept in the SoundFont.



Setup SoundFont and Midi files to play in your application

SoundFont available

GeneralUser_GS_SoftSynth_v144

Banks available in SoundFont GeneralUser_G:

Bank number 0	☑ Default bank	□ Drum Kit bank
Bank number 128	□ Default bank	☑ Drum Kit bank

Midi file available

MPTK_Example 0001
MPTK_Example 0002
MPTK_Example 0003
MPTK_Example 0004
MPTK_Example 0005
MPTK_Example 0006
MPTK_Example 0007

```
MPTK_Example 0004
Format: 1
Tracks: 6
Ticks Quarter Note: 96
[00000] [T:02 C:01] Piano
[00000] [T:03 C:01] Piano
[00000] [T:04 C:01] Clarinet
[00000] [T:05 C:01] Guitar
[00000] [T:06 C:01] Horn
[00000] [T:01 C:01] SetTempo Tempo:140 MicrosecondsPerQuarterNote:4285
[00000] [T:02 C:05] PatchChange 000 Acoustic Grand
[00000] [T:02 C:05] PatchChange 000 Acoustic Grand
[00000] [T:03 C:04] PatchChange 000 Acoustic Grand
```

Analyse Midi files and optimize SoundFont GeneralUser_G

☑ Keep all Zones ☑ Keep all Patchs

Patch count :	129
Wave count :	763
Wave size :	32 Mo

Integrate Midi Player in your script

Class MidiPlayerTK.MidiPlayerGlobal

Global class with static methods and properties.

Public Member Functions

- int **MPTK_FindMidi** (string name)
  Find a Midi file from name or partial name.
- float **MPTK_DistanceToListener**(Transform trf)
  Calculate distance with the AudioListener in your project.

Public Events

- UnityEvent **OnEventPresetLoaded**
  Event triggered when MPTK is ready to play.

Properties

- ReadOnlyCollection<string> **MPTK_ListMidis** [get]
  Get list of midi files available. It's a read only list. Add Midi file in edit mode with the menu tools of the Unity editor.

Class MidiPlayerTK.MidiFilePlayer

Play a selected midi file.

A set of Midi file must be defined with Midi Player Setup (menu tools) before.

Public Member Functions

- void **MPTK_Play** ()
  Play the midi file

- void **MPTK_Stop** ()
  Stop playing

- void **MPTK_RePlay** ()
  Restart playing the current midi file

- void **MPTK_Pause** ()
  Pause the current playing

- void **MPTK_Next** ()
  Play next Midi in list

- void **MPTK_Previous** ()
  Play previous Midi in list

Public Events

- UnityEvent **OnEventStartPlayMidi**
  Define unity event to trigger at start

- UnityEvent **OnEventEndPlayMidi**
  Define unity event to trigger at end. Not triggered when MPTK_Stop() is called.

Properties

- TimeSpan **MPTK_Duration** [get]
  Get duration of current Midi according the current tempo

- bool **MPTK_EnableChangeTempo** [get, set]
  Should change tempo from Midi Events ?

- bool **MPTK_IsPaused** [get]
  Is Midi file playing is paused ?

- bool **MPTK_IsPlaying** [get]
  Is Midi file is playing ?

- bool **MPTK_Loop** [get, set]
  Should automatically restart playing current Midi when Midi reach the end ?

- bool **MPTK_LogEvents** [get, set]
  Log midi events

- bool **MPTK_LogWaves** = false
  Log information on selected wave for each notes

- float **MPTK_MaxDistance** [get, set]
  MaxDistance to use for PauseOnDistance

- int **MPTK_MidiIndex** [get, set]
  Index Midi to play or playing. return -1 if not found

- string **MPTK_MidiName** [get, set]
  Midi name to play. Must exists in MidiPlayerGlobal.MPTK_ListMidis. List of Midi files must be defined with Midi Player Setup (menu tools) and contains that name.

- bool **MPTK_PauseOnDistance** [get, set]
  *Should the Midi playing must be paused if distance between AudioListener and **MidiFilePlayer** is greater than MaxDistance*

- bool **MPTK_PlayOnStart** [get, set]
  Should the Midi start playing when application start ?

- TimeSpan **MPTK_PlayTime** [get]
  Time from the start of playing the current midi

- float **MPTK_Position** [get, set]
  Position to play from 0 to lenght time of midi playing (in millisecond)

- float **MPTK_PulseLenght** [get]
  Lenght in millisecond of a quarter

- int **MPTK_Quantization** [get, set]
  Level of quantization : 0 = none to 5 = 64th Note

- float **MPTK_Speed** [get, set]
  Speed of playing. Between 0.1 (10%) to 5.0 (500%). Set to 1 for normal speed. Avoid changing when playing.

- double **MPTK_Tempo** [get]
  Get default tempo defined in Midi file or modified with Speed. Return QuarterPerMinuteValue similar to BPM (Beat Per Measure)

- int **MPTK_Transpose** [get, set]
  Transpose note from -24 to 24

- float **MPTK_Volume** [get, set]
  Volume of midi playing. Must be >=0 and <= 1

Have Fun !