

DJ PRO

Music is love



Documentation

crosstales LLC

Date: 14. October 2019

Version: 2019.4.1

Table of Contents

1. Overview.....	3
2. Features.....	4
2.1. DJ and turntables.....	4
2.2. Documentation & control.....	4
2.3. Compatibility.....	4
2.4. Supported third-party assets.....	5
3. Demonstration.....	6
3.1. Play.....	6
3.2. Mix.....	6
3.3. Piano.....	7
3.4. 3DAudio.....	7
3.5. Loudspeaker.....	8
3.6. PlaySingleFile.....	8
4. Setup.....	9
4.1. Schema.....	9
4.2. Turntable.....	10
4.3. Providers.....	11
4.3.1. RecordProviderPath.....	11
4.3.2. RecordProviderPlaylist.....	11
4.3.3. Other providers.....	11
4.4. Sets.....	12
4.4.1. RecordSet.....	12
4.4.2. DJSet.....	12
4.5. SimplePlayer.....	13
4.6. Other components.....	13
4.6.1. Loudspeaker.....	13
4.6.2. RecordSaver.....	13
4.6.3. CrossFader.....	13
4.6.4. RecordFader.....	13
4.6.5. Looper.....	13
4.6.6. SurviveSceneSwitch.....	14
4.6.7. Eventer.....	14
5. API.....	15
5.1. Callbacks.....	15
5.1.1. Playback start and end.....	15
5.1.2. Buffering start, end and progress.....	15
5.1.3. Audio start, end and time.....	16
5.1.4. Errors.....	16
5.1.5. Filter change.....	16
5.1.6. Records change.....	16
5.1.7. Record change.....	17
5.1.8. Example.....	17
5.2. Complete API.....	17
6. Third-party support (PlayMaker etc.).....	18
7. Verify installation.....	18
8. Upgrade to new version.....	18
9. Problems, improvements etc.....	18
10. Release notes.....	19
11. Credits.....	19
12. Contact and further information.....	20
13. Our other assets.....	21

Thank you for buying our asset "DJ PRO"!

If you have any questions about this asset, send us an email at dj@crosstales.com. Please don't forget to rate it or write a little review – it would be very much appreciated.

1. Overview

If you ever wanted to listen to your own music within your game, you can do so now. DJ is, simply put, the missing link between Unity and your music-library.

As the name promises, it's a solution to build the music player of your dreams – just like a real DJ. It adds and plays audio files from anywhere inside your application, it's very configurable and you can add as many records, sets and turntables as you wish.

DJ is a must for everyone who loves music!

2. Features

2.1. DJ and turntables

- Support for those **audio formats** on **any platform**:
 - MP3
 - OGG
 - XM
 - IT
 - MOD
 - S3M
 - WAV
 - AIFF
- **Export** audio from a **turntable** or the whole session
- Read the **lyrics** of the **current track**!
- Open **Spotify** with the **current track**!
- Tune into **multiple turntables** at the **same time** (and blend between records)
- Reads **tags** like **ID3**
- **Reads** and **saves** M3U, PLS and XSPF files
- **Crossfade** between **turntables** and **records**
- **Loop** between **start** and **end** position
- **Events** for certain positions
- **Cache tracks** to play them with no delay (e.g. a piano)
- **Watchdog** for file system changes
- **Performance**: Very low impact on performance!
- **No limits**: Does survive changing scenes! The music is not interrupted even during a load operation if necessary

2.2. Documentation & control

- **Test all audio tracks inside the editor!**
- Powerful [API](#) for **maximum control**!
- **Native file browser** for Windows and macOS!
- Detailed **demo scenes**!
- Comprehensive **documentation** and **support**!
- Full **C# source code**!

2.3. Compatibility

- Supports **all build platforms**
- Works with **Windows, Mac** and **Linux editors**!
- Compatible with **Unity 2017.4 – Unity 2019**
- Supports **AR** and **VR**!
- Works with [Online Check](#)
- Works with [File Browser](#)

2.4. Supported third-party assets

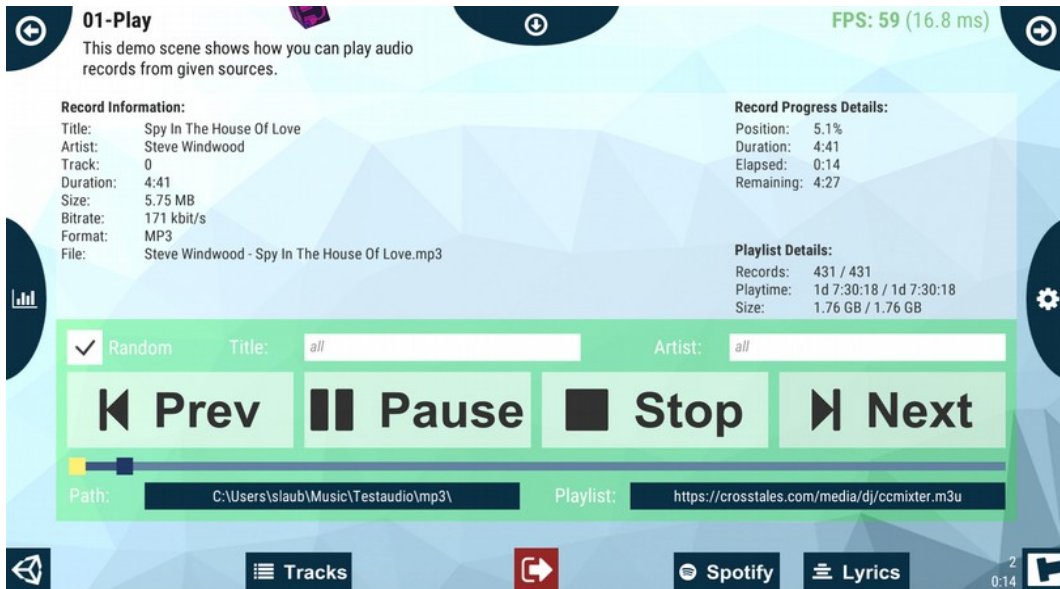
- [Audio Visualizer](#)
- [Complete Sound Suite](#)
- [PlayMaker](#)
- [Visualizer Studio](#)

3. Demonstration

The asset comes with many demo scenes to show the main usage.

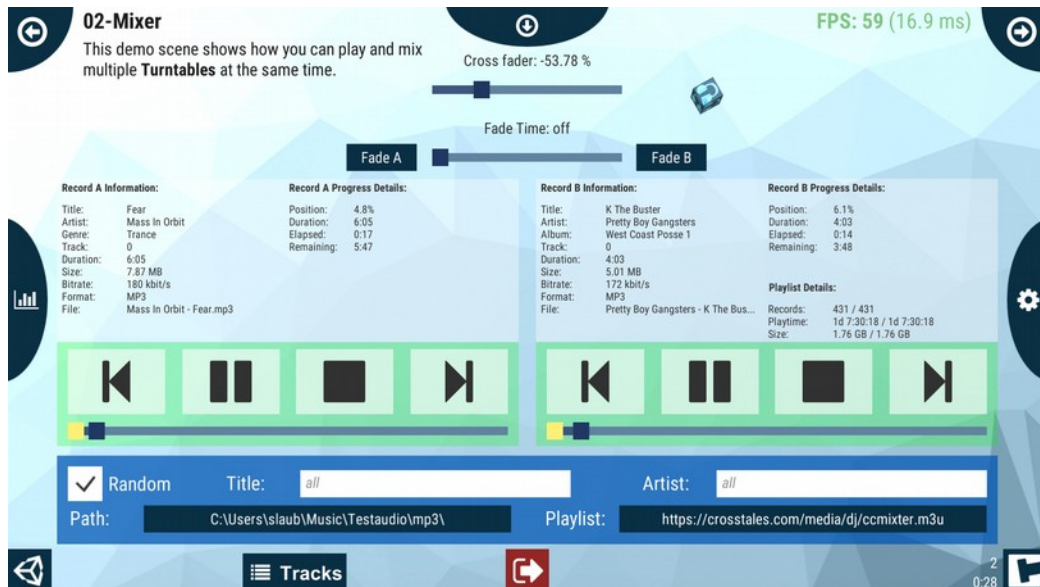
3.1. Play

This demo scene shows how you can play audio records from given sources.



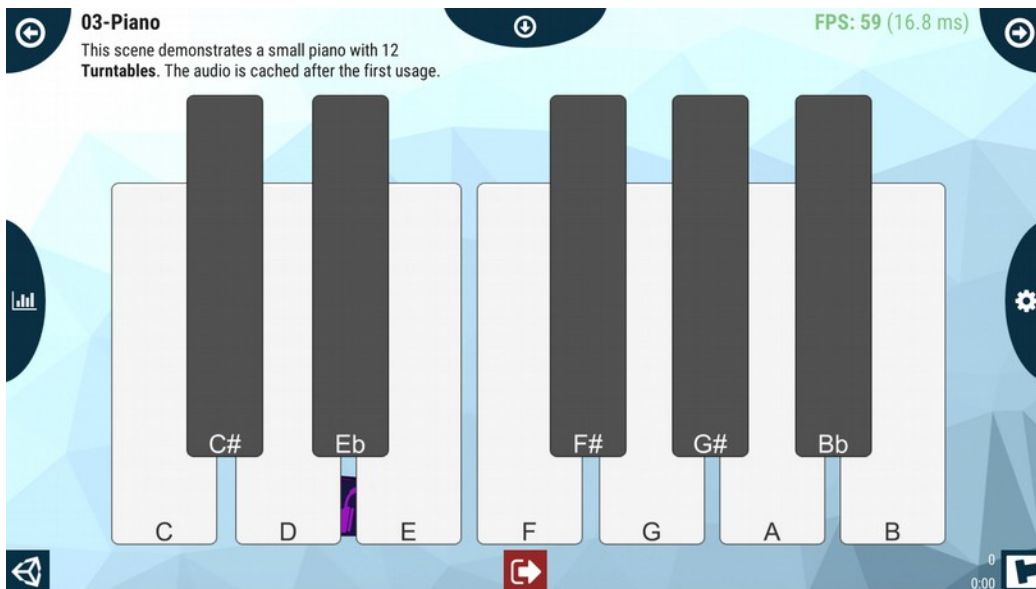
3.2. Mix

This demo scene shows how you can play and mix multiple **Turntables** at the same time.



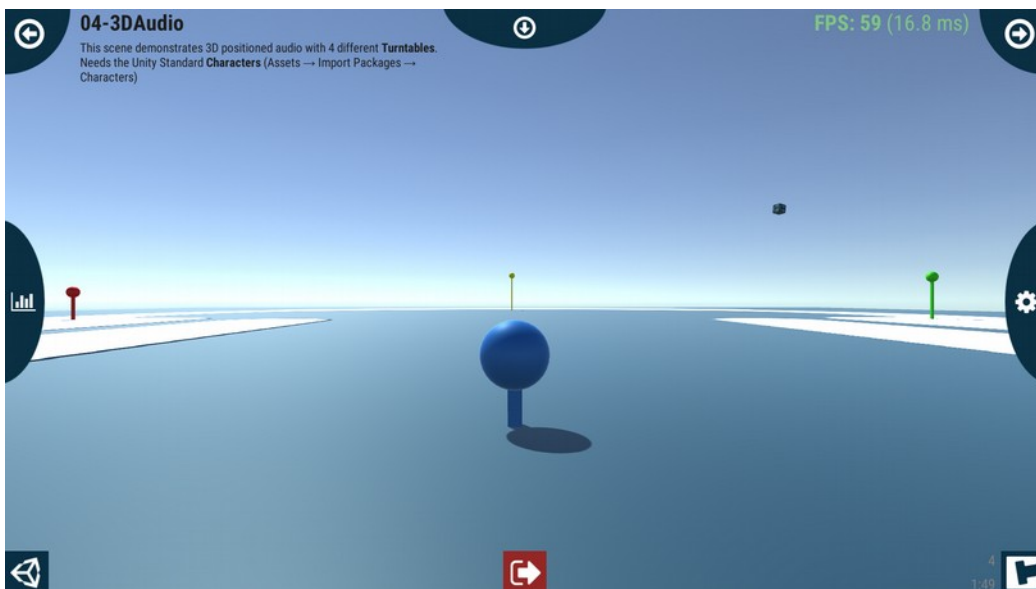
3.3. Piano

This scene demonstrates a small piano with 12 **Turntables**. The audio is cached after the first usage.



3.4. 3DAudio

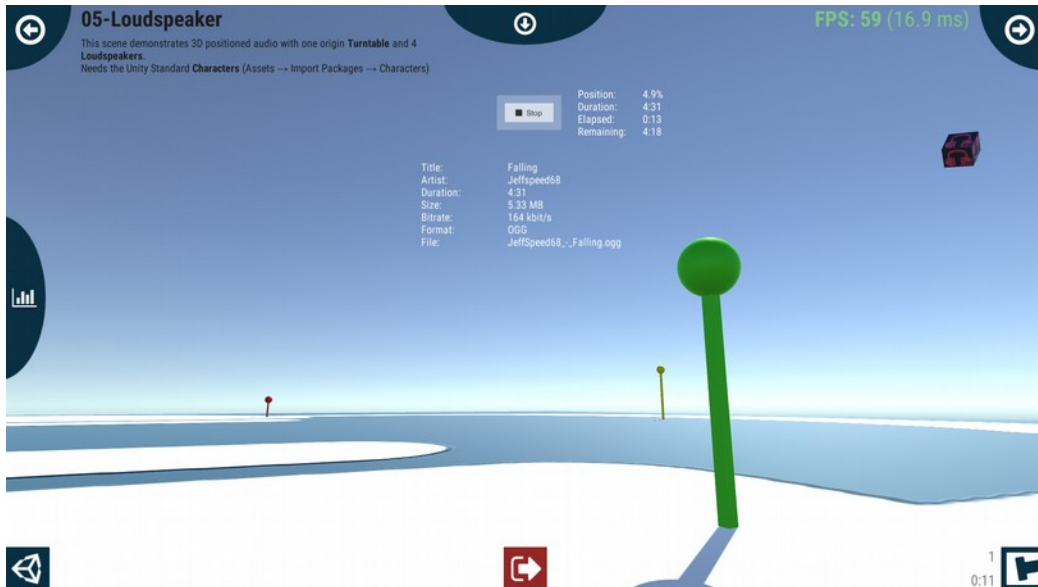
This scene demonstrates 3D positioned audio with 4 different **Turntables**.



Needs the [Unity Standard Assets](#)-package.

3.5. Loudspeaker

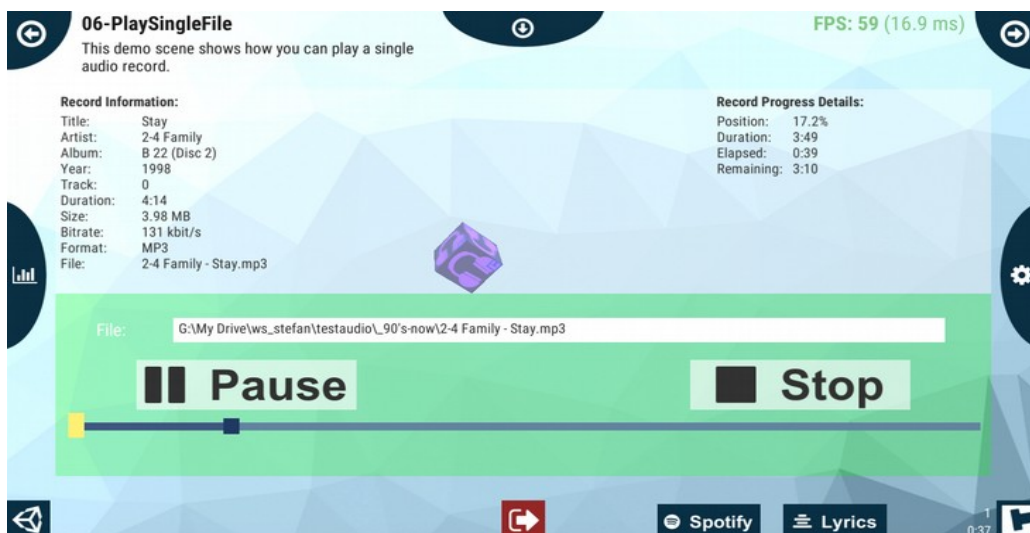
This scene demonstrates 3D positioned audio with one origin **Turntable** and 4 **Loudspeakers**.



Needs the [Unity Standard Assets](#)-package.

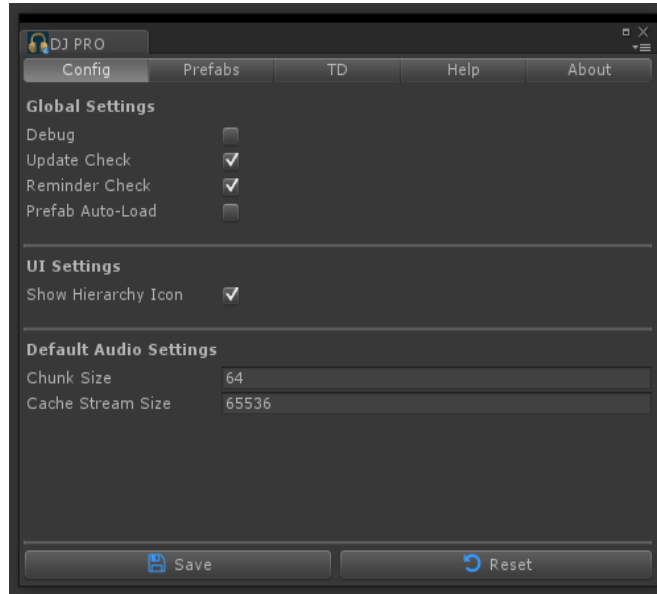
3.6. PlaySingleFile

This demo scene shows how you can play a single audio record with a Turntable.



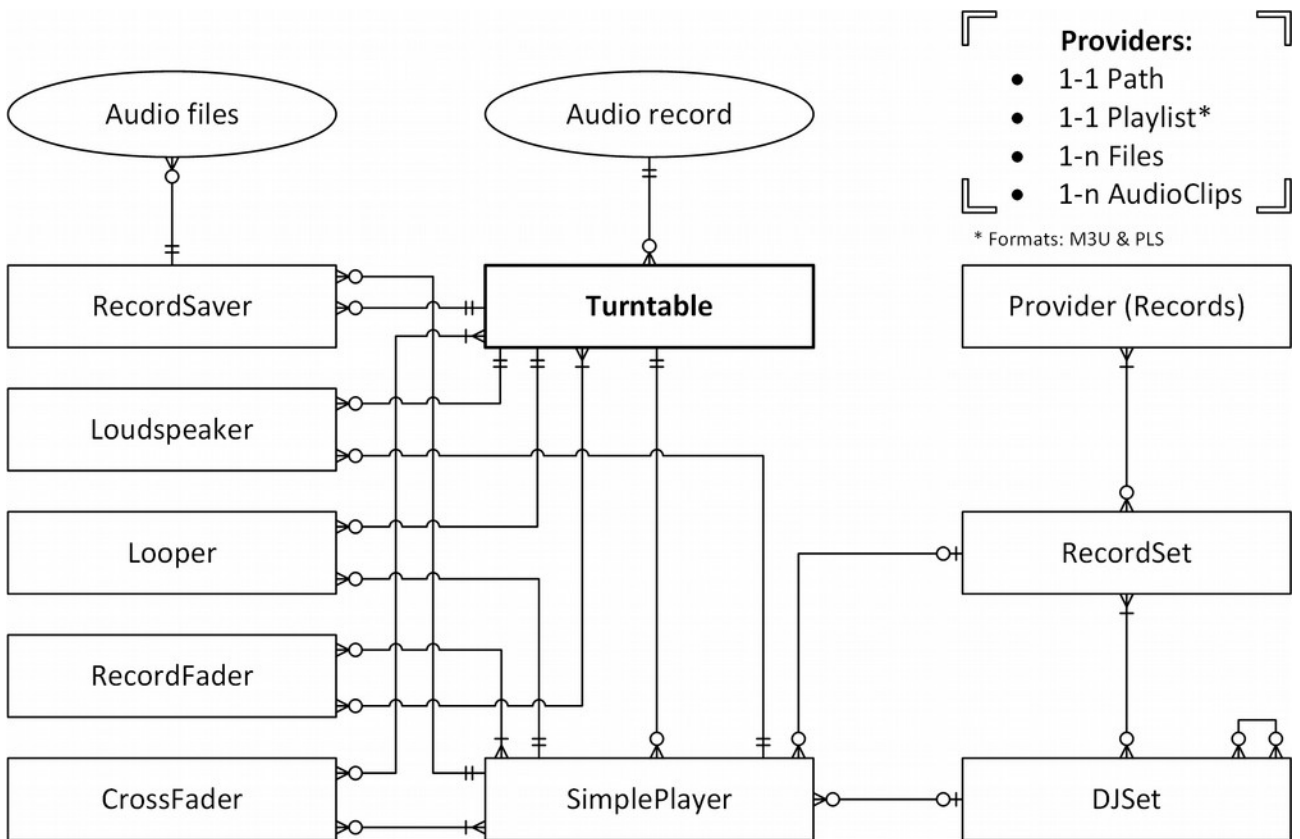
4. Setup

“DJ PRO” has global settings under “Edit\Preferences...” and under “Tools\DJ PRO\ Configuration...”:



4.1. Schema

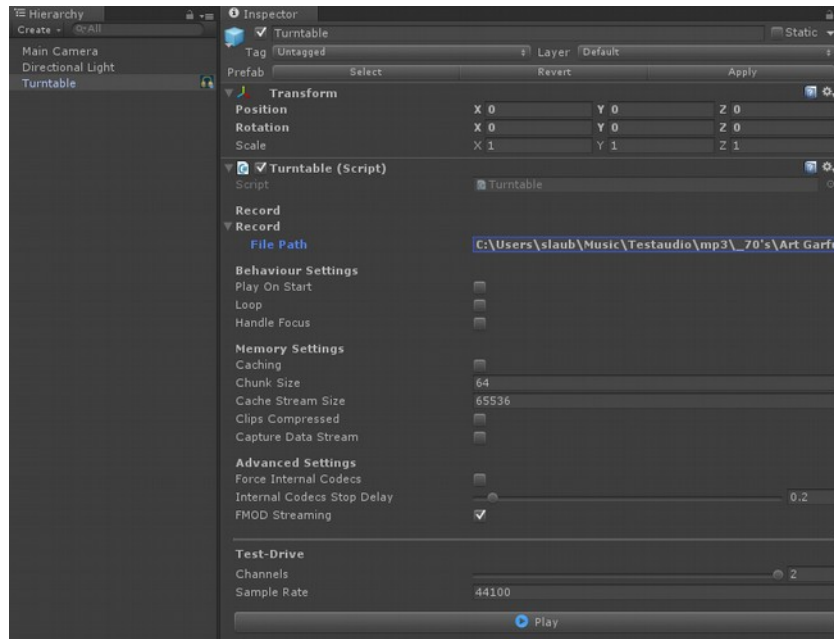
The following graphic explains the relationships between all relevant components:



4.2. Turntable

There are four ways to use a single turntable:

1. Add the prefab **Turntable** from Assets/Plugins/crosstales/DJ/Prefabs to the scene
2. Or go to *Tools => DJ PRO=> Prefabs => Turntable*
3. Right-click in the *hierarchy-window => DJ PRO => Turntable*
4. Add it from the Prefabs-tab.



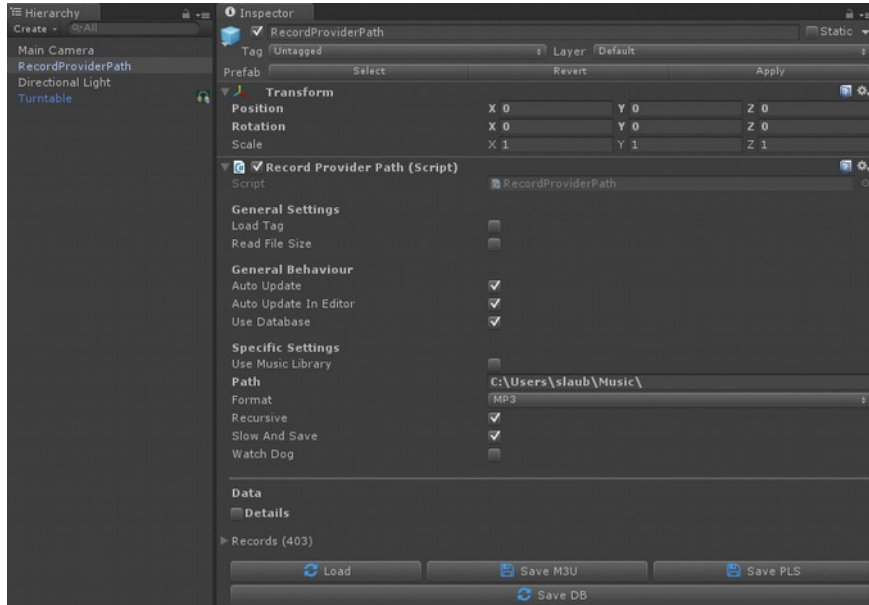
Just enter a file in "File Path" and you're good to go.

4.3. Providers

There are two different types of providers:

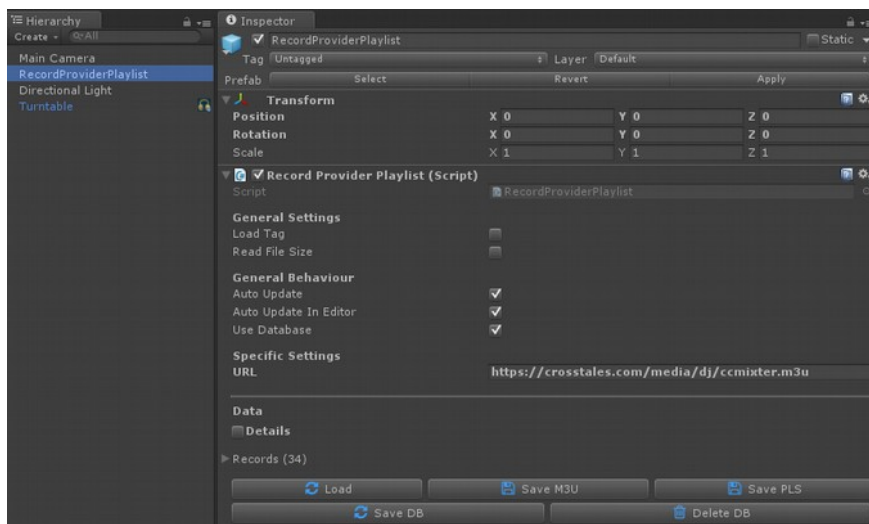
4.3.1. RecordProviderPath

This provider scans a path for audio files.



4.3.2. RecordProviderPlaylist

This provider reads a playlist file (M3U/PLS) with all audio files.



4.3.3. Other providers

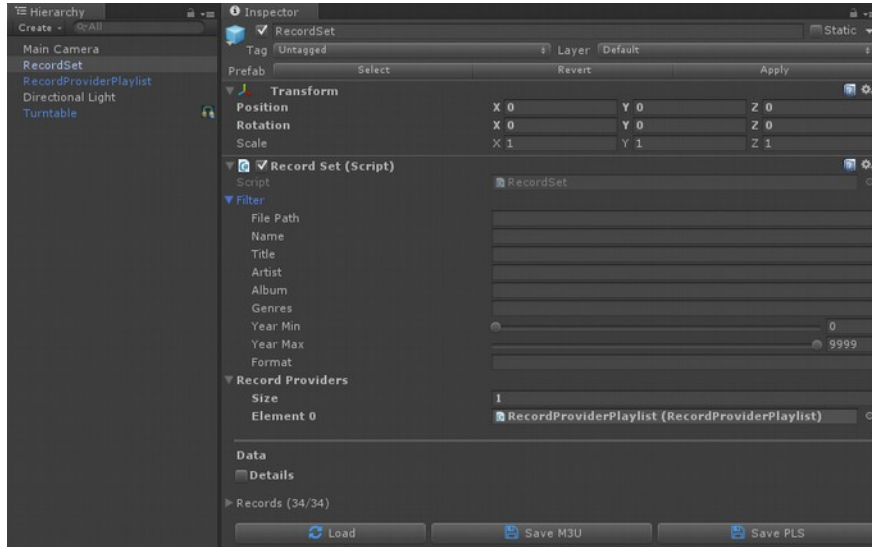
RecordProviderClip and RecordProviderFile manage a list of records.

4.4. Sets

There are two different types of sets:

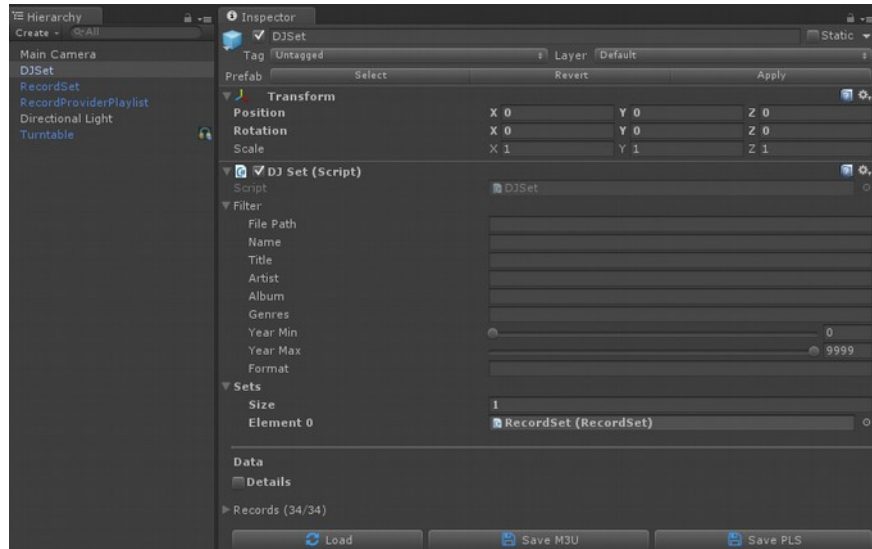
4.4.1. RecordSet

This set consists of 1-n providers.



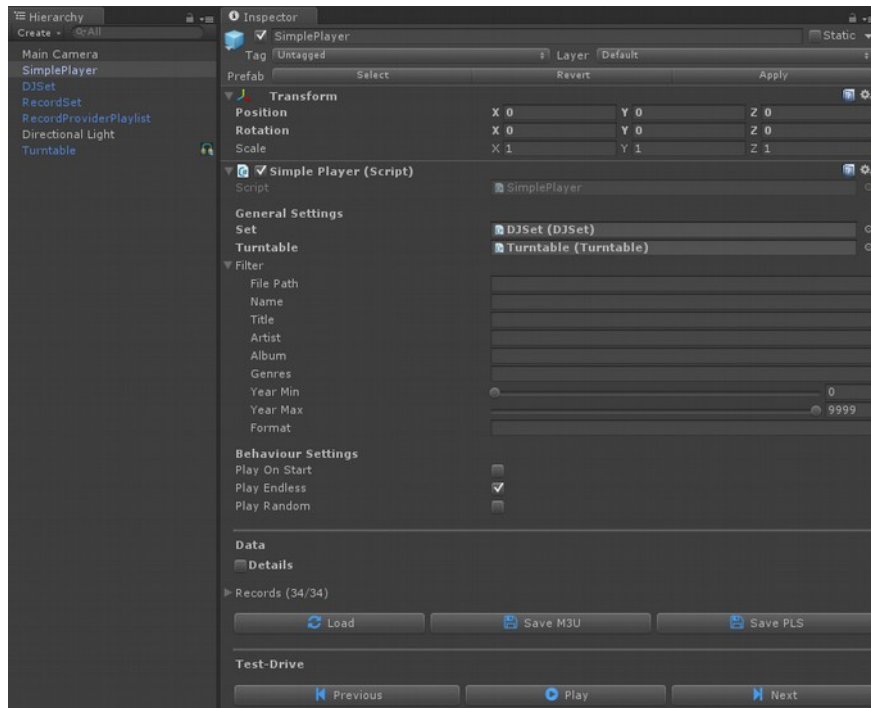
4.4.2. DJSet

This set consists of 1-n sets.



4.5. SimplePlayer

The "SimplePlayer" a „Turntable“ and a „Set“ and offers functions like „play next record“ etc.



4.6. Other components

The other components can be added in the same way as "Turntable"

4.6.1. Loudspeaker

This is useful to use the same player on multiple locations in the game.

4.6.2. RecordSaver

Allows to save songs from a player as WAV files.

4.6.3. CrossFader

Allows to fade between two players.

4.6.4. RecordFader

Allows to fade between records via two players.

4.6.5. Looper

Allows to set loop points in a record.

4.6.6. SurviveSceneSwitch

Allows any Unity gameobject to survive a scene switch. This is especially useful to keep the music playing while loading a new scene.

4.6.7. Eventer

Allows to set event points in a record.

5. API

The asset contains various methods and the most important are explained here. Make sure to **include** the **name space** in your relevant source files:

```
using Crosstales.DJ;
```

5.1. Callbacks

There are various callbacks available. Subscribe them in the "OnEnable"-method and unsubscribe in "OnDisable".

5.1.1. Playback start and end

```
PlaybackStart(Record record);  
PlaybackStart OnPlaybackStart;
```

Triggered whenever the playback starts.

```
PlaybackEnd(Record record);  
PlaybackEnd OnPlaybackEnd;
```

Triggered whenever the playback ends.

5.1.2. Buffering start, end and progress

```
BufferingStart(Record record);  
BufferingStart OnBufferingStart;
```

Triggered whenever the buffering starts.

```
BufferingEnd(Record record);  
BufferingEnd OnBufferingEnd;
```

Triggered whenever the buffering ends.

```
BufferingProgressUpdate(Record record, float progress);  
BufferingProgressUpdate OnBufferingProgressUpdate;
```

Triggered whenever the buffering progress changes.

5.1.3. Audio start, end and time

AudioStart(Record record);

AudioStart **OnAudioStart**;

Triggered whenever the audio starts.

AudioEnd(Record record);

AudioEnd **OnAudioEnd**;

Triggered whenever the audio ends.

AudioPlayTimeUpdate(Record record, float playtime);

AudioPlayTimeUpdate **OnAudioPlayTimeUpdate**;

Triggered whenever the audio playtime changes.

5.1.4. Errors

ErrorInfo(Record record, string info);

ErrorInfo **OnErrorInfo**;

Triggered whenever an error occurs.

5.1.5. Filter change

Callback for **Sets** and **SimplePlayer**.

FilterChange();

FilterChange **OnFilterChange**;

Triggered whenever a filter changes.

5.1.6. Records change

Callback for **Sets** and **SimplePlayer**.

RecordsChange();

RecordsChange **OnRecordsChange**;

Triggered whenever a the audio records change.

5.1.7. Record change

Callback for **SimplePlayer**.

```
RecordChange(Record record);  
RecordChange OnRecordChange;
```

Triggered whenever another record is played.

5.1.8. Example

```
void OnEnable() {  
    // Subscribe event listeners  
    Turntable.OnPlaybackStart += playBackStart;  
    Turntable.OnPlaybackEnd += playBackEnd;  
    Turntable.OnAudioPlayTimeUpdate += audioPlayTime;  
}  
  
void OnDisable() {  
    // Unsubscribe event listeners  
    Turntable.OnPlaybackStart -= playBackStart;  
    Turntable.OnPlaybackEnd -= playBackEnd;  
    Turntable.OnAudioPlayTimeUpdate -= audioPlayTime;  
}  
  
private void playBackStart(Record record) {  
    Debug.Log("Playback started");  
}  
  
private void playBackEnd(Record record) {  
    Debug.Log("Playback ended");  
}  
  
private void audioPlayTime(Record record, float playtime) {  
    Debug.Log("Playtime: " + Helper.FormatSecondsToHourMinSec(playtime));  
}
```

5.2. Complete API

Please read the [DJ-api.pdf](#) for more details.

6. Third-party support (PlayMaker etc.)

„DJ PRO“ supports various assets from other publishers. Please import the desired packages from the „3rd party“-folder.

7. Verify installation

Check if DJ is installed:

```
#if CT_DJ
    Debug.Log("DJ installed: " + Util.Constants.ASSET_VERSION);
#else
    Debug.LogWarning("DJ NOT installed!");
#endif
```

8. Upgrade to new version

Follow this steps to upgrade your version of "DJ PRO":

1. Update "DJ PRO" to the latest version from the "Unity AssetStore"
2. Inside your project in Unity, go to menu "File" => "New Scene"
3. Delete the "Assets/Plugins/crosstales/DJ" folder from the Project-view
4. Import the latest version from the "Unity AssetStore"

9. Problems, improvements etc.

If you encounter any problems with this asset, just [send us an email](#) with a problem description and the invoice number and we will try to solve it.

We will try and make a version for all platforms as well, please bear with us.

10. Release notes

See "VERSIONS.txt" under "Assets/Plugins/crosstales/DJ/Documentation" or online:
<https://crosstales.com/media/data/assets/DJ/VERSIONS.txt>

11. Credits

"DJ PRO" uses modified versions of:

- [NAudio 1.7.2](#)
- [NLayer 1.1.0](#)
- [NVorbis 0.8.4](#)
- [TagLib-sharp 2.1.0.0](#)

The search is provided by [Spotify](#).

The lyrics are provided by [AZLyrics](#).

The audio files for the demos are from [CC Mixer](#).

The icons are based on [Font Awesome](#).

We aren't affiliated to those companies.

12. Contact and further information

crosstales LLC

Schanzeneggstrasse 1

CH-8002 Zürich

Homepage: <https://www.crosstales.com/en/portfolio/DJ/>

Email: dj@crosstales.com

AssetStore: <https://assetstore.unity.com/lists/crosstales-42213>

Forum: <https://forum.unity.com/threads/coming-soon-dj-play-your-music.502627/>

Documentation: <https://www.crosstales.com/media/data/assets/DJ/DJ-doc.pdf>

API: <https://www.crosstales.com/media/data/assets/DJ/api/>










Windows-Demo: https://www.crosstales.com/media/data/assets/DJ/downloads/DJ_demo_win.zip

Mac-Demo: https://www.crosstales.com/media/data/assets/DJ/downloads/DJ_demo_mac.zip

Linux-Demo: https://www.crosstales.com/media/data/assets/DJ/downloads/DJ_demo_linux.zip

Android-Demo: <https://www.crosstales.com/media/dj/DJ.apk>

13. Our other assets

 <p>3D Skybox</p>	<p>Those beautiful packages contain professional 8k, HDR, stereoscopic 360° real-world skyboxes for your projects.</p>
 <p>Bad Word Filter</p>	<p>The "Bad Word Filter" (aka profanity or obscenity filter) is exactly what the title suggests: a tool to filter swearwords and other "bad sentences".</p>
 <p>File Browser</p>	<p>File Browser is a wrapper for native file dialogs on Windows, macOS, Linux and UWP (WSA).</p>
 <p>Online Check</p>	<p>You need a reliable solution to check for Internet availability? Here it is!</p>
 <p>Radio</p>	<p>Radio allows implementing free music from Internet radio stations into your project..</p>
 <p>RT-Voice</p>	<p>RT-Voice uses the computer's (already implemented) TTS (text-to-speech) voices to turn the written lines into speech and dialogue at run-time! Therefore, all text in your game/app can be spoken out loud to the player.</p>
 <p>True Random</p>	<p>True Random can generate "true random" numbers for you and your application. The randomness comes from atmospheric noise, which for many purposes is better than the pseudo-random number algorithms typically used in computer programs.</p>
 <p>Turbo Backup</p>	<p>Turbo Backup is the fastest and safest way to backup your Unity project. It only stores the difference between the last backup, this makes it incredible fast.</p>
 <p>Turbo Switch</p>	<p>Turbo Switch is a Unity editor extension to reduce the time for assets to import during platform switches. We measured speed improvements up to 100x faster than the built-in switch in Unity.</p>

