

File Browser

Native file browser for standalone



[Documentation](#)

crosstales LLC

Date: 14. January 2019

Version: 2019.1.0

Table of Contents

1. Overview.....	3
2. Features.....	3
3. Demonstration.....	4
3.1. ExampleSync.....	4
3.2. ExampleAsync.....	5
4. API.....	6
4.1. Open single file.....	6
4.2. Open multiple files.....	6
4.3. Open single folder.....	6
4.4. Open multiple folders.....	7
4.5. Save file.....	7
4.6. Complete API.....	7
5. Verify installation.....	7
6. Upgrade to new version.....	8
7. Problems, improvements etc.....	8
8. Release notes.....	8
9. Credits.....	8
10. Contact and further information.....	9
11. Our other assets.....	10

Thank you for using our asset "File Browser"!

If you have any questions about this asset, send an email to fb@crosstales.com.

Please don't forget to rate it or write a little review – it would be very much appreciated.

1. Overview

File Browser is a simple wrapper for native file dialogs on Windows, macOS and Linux.

2. Features

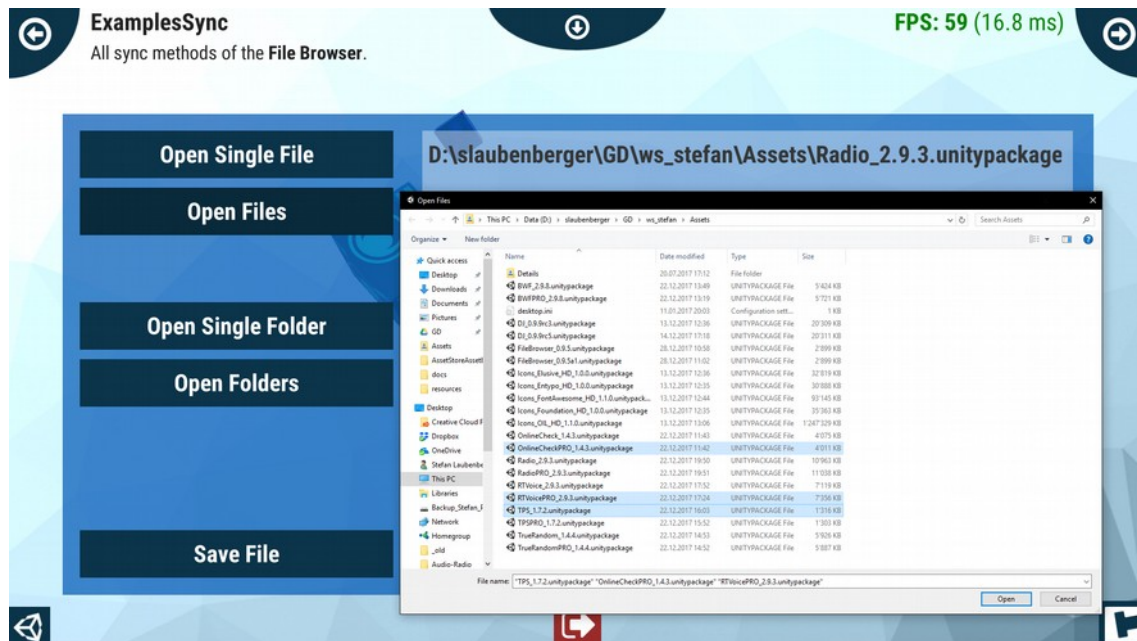
- Works with **Windows**, **macOS** and **Linux** in editor and runtime
- Open **file/folder**, **save file** dialogs supported
- **Multiple file** selection
- **Multiple folder** selection on **macOS** and **Linux**
- File extension **filters**
- Powerful [API](#) for **maximum control**!
- Detailed **demo scenes**!
- Comprehensive [documentation](#)!
- Compatible with **Unity 5.6 – Unity 2019**
- Full **C# source code**!

3. Demonstration

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

3.1. Demo

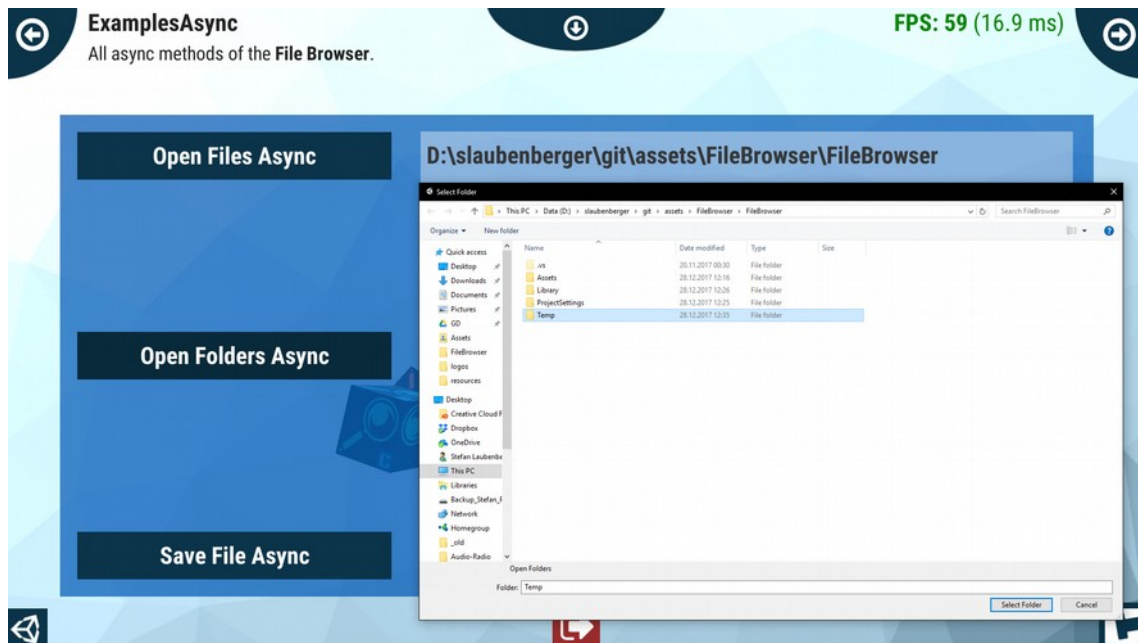
This scene shows open files/folders and a save dialogues.



3.2. ExampleAsync

This scene shows asynchronous open files/folders and a save dialogues.

Note: async methods are currently disabled



4. 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.FB;
```

4.1. Open single file

```
public void OpenSingleFile() {  
    string extensions = "";  
  
    string path = FileBrowser.OpenSingleFile("Open File", "", extensions);  
  
    Debug.Log("Selected file: " + path);  
}
```

4.2. Open multiple files

```
public void OpenFiles() {  
    string extensions = "";  
  
    string[] paths = FileBrowser.OpenFiles("Open Files", "", extensions, true);  
  
    foreach (string path in paths)  
    {  
        Debug.Log("Selected file: " + path);  
    }  
}
```

4.3. Open single folder

```
public void OpenSingleFolder() {  
    string path = FileBrowser.OpenSingleFolder("Open Folder");  
  
    Debug.Log("Selected folder: " + path);  
}
```

4.4. Open multiple folders

```
public void openFolders() {
    string[] paths = FileBrowser.OpenFolders("Open Folders");

    foreach (string path in paths)
    {
        Debug.Log("Selected folder: " + path);
    }
}
```

Note: this works only under macOS

4.5. Save file

```
public void SaveFile() {
    string extensions = "txt";

    string path = FileBrowser.SaveFile("Save File", "", "MyFile", extensions);

    Debug.Log("Save file: " + path);
}
```

4.6. Complete API

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

5. Verify installation

Check if File Browser is installed:

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

6. Upgrade to new version

Follow this steps to upgrade your version of "File Browser":

1. Update "File Browser" to the latest version from the "Unity AssetStore"
2. Delete the "Assets/Plugins/crosstales/FileBrowser" folder from the Project-view
3. Import the latest version downloaded from the "Unity AssetStore"

7. Problems, improvements etc.

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

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

8. Release notes

See "VERSIONS.txt" under "Assets/Plugins/crosstales/FileBrowser/Documentation".

9. Credits

Partially based on:

<https://github.com/gkngkc/UnityStandaloneFileBrowser>

Windows dialogs:

<http://www.ookii.org/software/dialogs/>

10. Contact and further information

crosstales LLC

Schanzeneggstrasse 1

CH-8002 Zürich

Homepage: <https://www.crosstales.com/>

Email: fb@crosstales.com

AssetStore: <https://goo.gl/qwtXyb>

Forum: <https://forum.unity.com/threads/file-browser-native-file-browser-for-windows-and-macos.510403/>

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










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

Windows-Demo: https://www.crosstales.com/media/data/assets/FileBrowser/downloads/FileBrowser_win.zip

Mac-Demo: https://www.crosstales.com/media/data/assets/FileBrowser/downloads/FileBrowser_mac.zip

Linux-Demo: https://www.crosstales.com/media/data/assets/FileBrowser/downloads/FileBrowser_linux.zip

11. Our other assets

 <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>DJ</p>	<p>DJ is a player for external music-files. It allows a user to play his own sound inside any Unity-app. It can also read ID3-tags.</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>RSockpol</p>	<p>Reliable Socket Policy Server which acts as replacement for Unitys own „sockpol.exe“.</p>
 <p>RTVoice</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>TPS</p>	<p>Turbo Platform 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>
 <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>

