QWinFF Documentation

Release 0.2.0

Timothy Lin <lzh9102@gmail.com>

September 23, 2013

ONE

INTRODUCTION

QWinFF is a GUI for FFmpeg [1], a command line media converter. FFmpeg can read audio and video files in various formats and convert them into other formats. For example, you can convert video downloaded from the Internet into audio files for your mobile phone.

QWinFF comes with a set of predefined presets to help you quickly convert media files without having to know the details. These presets defines what arguments should be passed to *ffmpeg* for each output format. In case you want to adjust the details (bitrate, audio volume, etc.), QWinFF lets you edit conversion parameters for each individual task.

Currently QWinFF runs on GNU/Linux and Microsoft Windows [2].

---

[1] http://www.ffmpeg.org/
[2] Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

USING QWINFF

2.1 QuickStart

When you start QWinFF, you will see the **Main Window**. At the top of the Main Window is the **Toolbar**, which contains most common operations you will need. Below the Toolbar is the Task List. The Task List is used to display all conversion tasks. At the bottom-right corner, there is a **Start** button. At the bottom of the window is the **Status Bar**.

The first thing to do is add some files to convert. You can do it by either clicking on the **Add** button on the Toolbar or drag-and-drop files into the Task List. After adding all files you want to convert, **Add Tasks Dialog** will appear. You can select the output container format (**Convert To**) and the output configuration (**Preset**). Then, select the desired output folder and press **Finish**.

Note: You can select multiple files in the **Open File** dialog and add them all at once.

When you have added some tasks, click the Start button and the conversion will start. Conversion progress of each task will be shown in the progress bar entry in the Task List. After a task has finished, double-click on it to open the output folder. If a conversion fails, you can view the error message by moving the mouse pointer over the progress bar or right-click on the task and select **Show Error Message**.

2.2 Task List

The task list shows all conversion tasks. Each task has some attributes like *Source*, *Destination*, *Duration* and *Progress* shown in different columns. You can reorder the columns by dragging the header items. Also it is possible to select which columns to display by right-clicking on the header of the task list. Available columns and their meaning are described as follows:

Source name of the input file

Destination name of the output file

Duration audio/video duration of the input file

Progress a progressbar showing the conversion progress

File Size size of the input file

Sample Rate audio sample rate (Hz) of the input file

Audio Bitrate audio bitrate (kb/s) of the input file

Channels the number of audio channels in the input file

Audio Codec audio codec name of the input file

Dimensions video width and height of the input file

Video Bitrate video bitrate (kb/s) of the input file

Framerate video frame rate (fps) of the input file

Video Codec video codec name of the input file

2.3 Add Tasks Dialog

The first section of the Add Tasks dialog is a list of files to be added. You can click on the **Add** button to add more files or click on the **Remove** button to remove files. When you are finished, click **Next** to enter the next section.

The second section contains settings related to the output file.

Convert to This option contains a drop-down list with all possible output extensions. This is the *container format* of the output file.

Preset For each extension, there are one or more presets that defines the *codec* and parameters used to convert the file.

Output Path Select the output folder of the tasks. This setting only applies to the current tasks. You can select different output folder when you add other tasks later.

2.3.1 Output Path Settings

New in version 0.1.9.

There are 3 choices of how the output folder is generated.

The first choice is **Select Folder**, which means you can assign a fixed output folder. If you select this one, the tasks will be output to the folder you choose.

The second choice is **Create new folder in source folder**. If you select this option, a new folder will be created in the parent folder of each input file. Each output file will be output to the new folder for its input file.

The third choice is **Output to source folder**. This option is similar to the second one, except it doesn't create a new folder. Each output file is placed in the same directory as its input file.

QWinFF automatically renames a output file if a file with the same filename already exists in the output folder. You don't have to worry about overwriting existing files.

FILES

3.1 Installation Directory

Installed files are placed differently on different platforms. On GNU/Linux, the program executable is installed in `<prefix>/bin` and data files are installed in `<prefix>/share/qwinff`. The `<prefix>` part depends on the packager, and the most common prefix is `/usr`. On Microsoft Windows, all files, including the executable file and data files, are placed in the installation directory you choose, and the default is `C:\Program Files (x86)\QWinFF` or `C:\Program Files\QWinFF`.

3.2 Data Files

The contents of the data directory are described as follows:

presets.xml This file contains ffmpeg presets. See *Presets* for details about the preset file format.

constants.xml This file contains some pre-defined values used by the program. It is read once on program startup. The function of each entry is documented using xml comments in the file. QWinFF won't start if errors or incorrect values are detected in this files, so modify this file with care.

translations/ This directory contains translation files (`*.qm`) for the program. Translation files should be named as `qwinff_<language-code>_<country-code>_.qm`. For example, translation file for Chinese (Taiwan) should be named `qwinff_zh_TW.qm`.

***.dll** (Windows only) Dynamic libraries the program depends on.

tools/ (Windows only) This directory contains external tools used by the program, including `ffmpeg.exe`, `ffprobe.exe`, `sox.exe` and other supporting tools/libraries.

3.3 Settings File

Program settings are saved separately for each user. On GNU/Linux, the settings file is `~/.qwinff/qwinff.ini`. The installed version on Windows saves the settings in `.qwinff\qwinff.ini` in the user's home directory, while the portable version saves the settings in `qwinff.ini` in the program directory.

PRESETS

Presets of QWinFF are written in *presets.xml* in the program installation directory. The format and structure of *presets.xml* is compatible with the preset file of WinFF [1]. This means you can copy any WinFF preset to *presets.xml* and use it in QWinFF. In fact, most QWinFF presets are taken from WinFF with little or no modification.

A preset contains the following fields:

label a descriptive name of the preset. example: `WebM Generic`

params parameters to be passed to ffmpeg. example: `-vcodec libvpx -acodec libvorbis -aq 90 -ac 2`

extension extension of the output file. example: `webm`

category category of the preset. This field is reserved but currently unused by QWinFF. example: `Websites`

The preset described above is then written in the preset file like this:

```
<WebMGeneric>
  <label>WebM Generic</label>
  <params>-vcodec libvpx -acodec libvorbis -aq 90 -ac 2</params>
  <extension>webm</extension>
  <category>Websites</category>
</WebMGeneric>
```

Note: The preset tag name (`WebMGeneric`) is chosen arbitrarily. Also, there is no need to make preset tag names unique. For example, it's OK to name all of them `preset`.

It is better to include codec information in `params` (e.g. `-vcodec libvpx`), because QWinFF checks whether the installed ffmpeg supports the codec. If any one of the `-acodec` or `-vcodec` isn't supported, the preset will be hidden.

---

[1] http://winff.org

CREDIT AND LICENSE

Copyright (C) 2013 by Timothy Lin, all rights reserved.

Related Artwork and the QWinFF Logo are designed by kuanyui.

QWinFF is heavily influenced by WinFF [1], another ffmpeg GUI frontend. Most of the presets used in QWinFF are directly copied from WinFF presets. The name *QWinFF* also comes from the notion that it is a program similar to WinFF and written in Qt.

QWinFF is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 or 3 of the License.

QWinFF is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

This document is licenced under the GNU Free Documentation License, either version 1.3 or any later version published by the Free Software Foundation.

FFmpeg is a separate project, not connected to QWinFF. FFmpeg is licensed under the LGPL version 2.1, and some optional parts of it are covered by the GPL. *FFmpeg* is a trademark of Fabrice Bellard, originator of the FFmpeg project.

SoX [2], the Swiss Army knife of audio manipulation, is used by QWinFF to process audio under some conditions. SoX is licensed under the GPL; either version 2, or any later version.

---

[1] http://winff.org
[2] http://sox.sourceforge.net