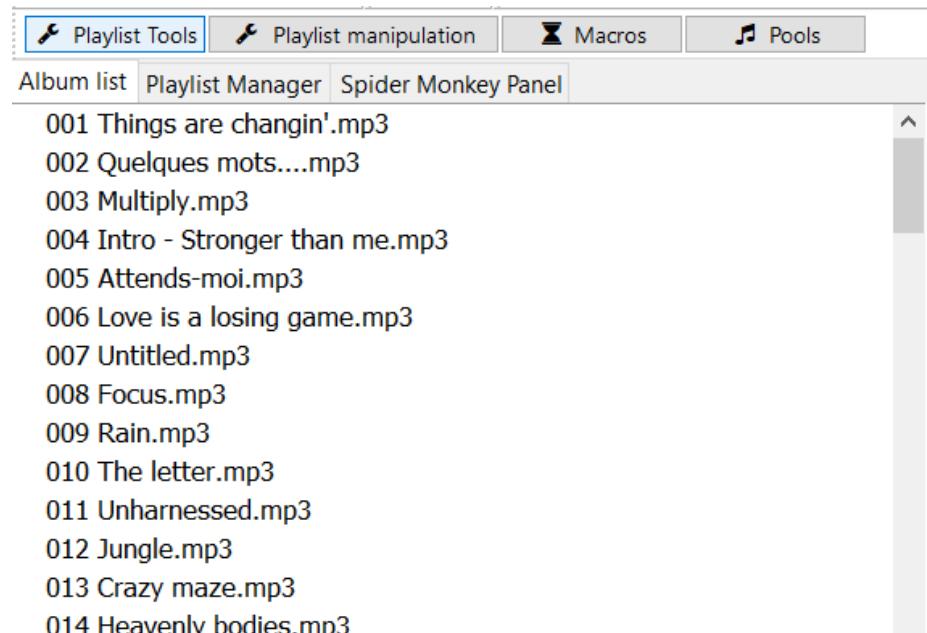


# Playlist Tools

A SWISS-KNIFE TOOL FOR FOOBAR2000  
USING SPIDER MONKEY PANEL AND JAVASCRIPT

BY  
**REGORXXX**

AUTHOR OF "PLAYLIST-MANAGER-SMP", "WORLD-MAP-SMP", ETC.



The screenshot shows a software interface titled 'Playlist Tools'. At the top, there is a navigation bar with tabs: 'Playlist Tools' (selected), 'Playlist manipulation', 'Macros', and 'Pools'. Below the navigation bar, there is a sub-navigation bar with tabs: 'Album list', 'Playlist Manager' (selected), and 'Spider Monkey Panel'. The main area displays a list of 14 MP3 files, each with a small thumbnail icon and a file name. The files are listed vertically:

- 001 Things are changin'.mp3
- 002 Quelques mots....mp3
- 003 Multiply.mp3
- 004 Intro - Stronger than me.mp3
- 005 Attends-moi.mp3
- 006 Love is a losing game.mp3
- 007 Untitled.mp3
- 008 Focus.mp3
- 009 Rain.mp3
- 010 The letter.mp3
- 011 Unharnessed.mp3
- 012 Jungle.mp3
- 013 Crazy maze.mp3
- 014 Heavenly bodies.mn3

Playlist-Tools-SMP  
Published on GitHub  
January 19, 2023

© 2023 regorxxx  
Contact me at: regorxxx@protonmail.com

**This publication should be colorblind-friendly; please, contact me if you find any accessibility difficulty or open an issue at GitHub.**

This program is free software: you can redistribute it and/or modify it under the terms of the GNU Afferro General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program 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 Afferro General Public License for more details.

You should have received a copy of the GNU Afferro General Public License [XI] along with this program. If not, see <https://www.gnu.org/licenses/>.

# Contents

<b>I Introduction</b>	<b>6</b>
<b>1 Why do we need more tools with so many components already available?</b>	<b>6</b>
1.1 Full fledged HTTP controller integration . . . . .	6
1.2 Expand Foobar2000 queries: dynamic queries . . . . .	7
1.3 Smart duplication removal . . . . .	8
1.4 Playlists and pools: auto-playlists are thing of the past . . . . .	8
1.5 Expand Foobar2000 automation: macros and configurable menus . . . . .	9
1.6 FLS slots limit (DLL's): Reduce components loaded with better replacements . . . . .	9
<b>II Installation</b>	<b>10</b>
<b>2 Scripts files and dependencies</b>	<b>10</b>
<b>3 Portable installations tip</b>	<b>10</b>
<b>4 Installation within Foobar2000's UI</b>	<b>11</b>
<b>5 Installation as a toolbar</b>	<b>12</b>
5.1 Advantages over standard panels . . . . .	14
<b>6 Installation along Buttons framework compatible scripts</b>	<b>14</b>
<b>7 Wine installations</b>	<b>15</b>
<b>8 Updating from older script versions</b>	<b>16</b>
<b>9 Nightly versions</b>	<b>16</b>
<b>III Buttons and Toolbar framework</b>	<b>17</b>

<b>10</b>	<b>Introduction</b>	<b>17</b>
<b>11</b>	<b>Loading preset</b>	<b>18</b>
<b>12</b>	<b>Adding buttons</b>	<b>19</b>
<b>13</b>	<b>Removing buttons</b>	<b>20</b>
<b>14</b>	<b>Changing buttons position</b>	<b>21</b>
<b>15</b>	<b>Toolbar orientation</b>	<b>22</b>
<b>16</b>	<b>Reflow buttons</b>	<b>23</b>
<b>17</b>	<b>Icons-only mode</b>	<b>24</b>
<b>IV</b>	<b>UI</b>	<b>25</b>
<b>18</b>	<b>Features</b>	<b>25</b>
18.1	Tooltip . . . . .	25
<b>19</b>	<b>Customization</b>	<b>26</b>
19.1	Custom color . . . . .	26
<b>V</b>	<b>Other scripts integration</b>	<b>27</b>
<b>VI</b>	<b>FAQ</b>	<b>28</b>
<b>VII</b>	<b>Advanced tips</b>	<b>29</b>
<b>20</b>	<b>Example</b>	<b>29</b>
<b>21</b>	<b>Portable 'plug&amp;play' installation</b>	<b>30</b>

<b>22 Pools using Playlist-Manager-SMP</b>	<b>31</b>
<b>VIII Technical notes</b>	<b>33</b>
<b>IX Known problems</b>	<b>34</b>
<b>X Bug procedure</b>	<b>35</b>
<b>XI License</b>	<b>36</b>

## Part I

# Introduction

## 1 Why do we need more tools with so many components already available?

### 1.1 Full fledged HTTP controller integration

There are multiple Foobar2000 controllers out there, like foo\_httpcontrol. Obviously, they are limited to the features integrated on the component so many advanced tools are out of question. SMP integration could bypass that limitation since it essentially allows to work around almost any Foobar2000 feature in multiple contexts (playlists, library, etc.), even the file system.

Regrettably, SMP is not integrated into any of those HTTP controllers... but that can be remediated with the use of 'ajquery-xxx' controller. Since the component allows to call arbitrary main menu entries, the 9 hardcoded menus by SMP can be called with the online controller when using the appropriate template.

The interesting part comes when using Playlist Tools along the controller, since the scripts can dynamically set those menus, provide names and descriptions, icons, allow output device and DSP selection, etc. Essentially full server control or running any of these tools from any device.

## 1.2 Expand Foobar2000 queries: dynamic queries

Ever wanted to search tracks with same artist than the current track? Some people may know foo\_quicksearch, which lets you predefine some tags to look for. Problem? They don't work with multi-value tags (order changes results), neither with multiple selections nor are integrated with queries.

Lets talk about Dynamic Queries: queries which adapt to the currently selected track. i.e. placeholders tags are substituted with the actual values of the currently selected track (expanded in case multi-values are present), then the query is evaluated as usual. Queries created this way are pretty situational, save a lot of writing time and are meant to be used by multiple playlist creation tools.

While 'foo\_quicksearch' may come handy for easy lookups, like searching for 'Same artist' than one track, Dynamic queries allow you to look for things like 'Same artist with different date' or 'Tracks which contain the selected artist as composer'. As can be seen, the sky is the limit.

ARTIST IS #ARTIST# AND NOT DATE IS #DATE#

COMPOSER IS #ARTIST#

Also they are fully compatible with multiple selections. If current selection contains multiple tracks by artists 'Bob Dylan' and 'Pedro Pastor', they would expand oto

(COMPOSER IS Bob Dylan) OR (COMPOSER IS Pedro Pastor)

(ARTIST IS Bob Dylan AND NOT DATE IS 1970) OR (ARTIST IS Pedro Pastor AND NOT DATE IS 2016)

And a track with multiple artists:

(ARTIST IS Bob Dylan AND ARTIST IS Johnny Cash) AND NOT DATE IS 1970

Note this is totally different to what other quick-search tools do. Because simply matching all tag values enumerated with commas leads to errors when values are not sorted the same.

%ARTIST% IS Bob Dylan, Johnny Cash

Does not retrieve the same results than

%ARTIST% IS Johnny Cash, Bob Dylan

But the next one covers all cases (even if the track has 3+ artists...).

ARTIST IS Bob Dylan AND ARTIST IS Johnny Cash

Also when one set of tags contains another one but also include extra values:

%Mood% IS Acoustic, Sad

Is different to the next one, which would also include tracks with additional mood values.

Mood IS Acoustic AND Mood IS Sad

Imagine using this tool in 10 artists a few times; how much it would take manually on different set of artists and values?

### 1.3 Smart duplication removal

Did you ever create an auto-playlist with all Rock tracks from the 60s only to found later that you have 4 versions of the same track by Hendrix? Well I do, many times in many use cases. And usually I don't want to have the same track repeated in a playlist... According to Foobar2000, duplicates are only physical files which match. i.e. adding the same exact file 2 times to a playlist.

Did you ever tried to remove those duplicates by hand? Good luck in a playlist with thousand of tracks. This set of tools provide multiple ways to remove (and filter) playlist or selection using TF and RegExp. For ex: finding duplicates by same ARTIST—DATE—TITLE. It doesn't matter if they come from different album, paths, are in mono or any format...

### 1.4 Playlists and pools: auto-playlists are thing of the past

The script includes tools for playlist creation similar to Random Pools component. Multiple playlists \library sources (pools) can be set to fill a destination playlist.

Configurable selection length per source, query filtering, picking method (random, from start, from end) and final sorting of destination playlist. They may even use dynamic queries changing the way the pools behave according to selection (for ex. a pool which outputs tracks with same key than selected track + another one which outputs same genre tracks), one of the main limitations of Random Pools component (which would require to create a pool for every desired genre for ex). Duplicates are also removed (see above).

Not only the library or playlists may be used as source but also the output of other playlist generation tools, like Search-by-Distance-SMP, to create smart mixes. Final playlist can also be sorted with special rules, like harmonic mixing.

## **1.5 Expand Foobar2000 automation: macros and configurable menus**

Macros feature allow to record and save the menus entries used, as a macro, to be called later. Automatic custom playlist creation and edits without limits. It works with all tools (only limitation are popups, which still require user input).

Many tools allow you to add your own presets (for ex. Standard Queries) as menu entries for later use. They may be used along macros to greatly expand their functionality, exported and imported as "addons".

Menus and tools can be disabled/enabled on demand, associated to keyboard shortcut which work globally within Foobar2000 context (so they are essentially equivalent to native menus), and much more...

## **1.6 FLS slots limit (DLL's): Reduce components loaded with better replacements**

One of the main limitations of windows (and thus Foobar2000) is the maximum number of plugins (DLL's) that can be associated to a given process. Thus, in some installations, specially those using VSTs, when the limit is reached strange things start happening: random crashes, bugs, etc. Something I have experienced myself while running or installing a few VSTs as DSPs within the software.

It's not so hard to reach that limit since many components use multiple DLL's! When you count the ones by foobar itself, VSTs, etc. as soon as you configure a bit your installation you come into problems. Therefore, Playlist Tools is a solution that can help in that sense, replacing multiple components whose functionality is not only included but also improved: Random Pools, Playlist Revive, Best version picker, Database Search, ...

## Part II

# Installation

## 2 Scripts files and dependencies

**Spider Monkey Panel 1.5.2 or higher is required.** Only stable releases are supported.

Copy all files from the zip into 'YOUR\_FOOBAR\_PROFILE\_PATH\scripts\SMP\xxx-scripts: If the folders don't exist, create them. Any other path WILL NOT work without editing the scripts. (see '\_TIPS and INSTALLATION\_vX.X.jpg'). Multiple scripts may share some files (specially helpers) so overwrite if asked to do so. Then load the script named '**buttons\_toolbars**' into a SMP panel within foobar [4].

- For **standard installations:** 'C:\Users\yourUser\AppData\Roaming\foobar2000\scripts\SMP\xxx-scripts\...'
- For **portable installations >= 1.6:** '.\foobar2000\profile\scripts\SMP\xxx-scripts\...'
- For **portable installations <= 1.5:** '.\foobar2000\scripts\SMP\xxx-scripts:.'

If you upgraded to >1.6 from an older portable version then it may be possible that the 'profile' folder does not exist. In such case you have to create it and move all the configuration folders/files to it, where they should reside (instead of the root of foobar2000 installation path). If you don't move all the configuration folders/files, then, on startup, default values will be used for things not found, probably "losing" the theme or other customization. You may "fix" it later moving the missing files which still reside in the root. May take some tries to do them all.

- **Some native folders and files which must be moved include:** index-data, js.data, component-updates, configuration, crash reports, user-components, foo\_spider\_monkey\_panel, library, playlists, theme.fth, LargeFieldsConfig.txt, version.txt
- **Some extra folders from other components which must be moved include (non extensive list):** autobackup, dvda\_metabase, foo\_httpcontrol\_data, foo\_youtube, images, lastfm, python, sacd\_metabase, vst-presets, yttm, minibar.db, playlist-tree-0.pts, playlist-tree-1.pts

Additionally, some fonts are required on the system. They can be installed by double clicking on the '.ttf' files and then selecting the 'install' option (requires Admin rights):

- **Font Awesome:** found at '.\resources\fontawesome-webfont.ttf'

## 3 Portable installations tip

When the script finds it's being loaded within a portable installation, it will set the default paths using relative paths. It will also warn with popups and/or the console about the -non recommended- use of absolute paths on portable installations. For more info see [21].

## 4 Installation within Foobar2000's UI

Once all files have been installed to the right paths, along their dependencies, an Spider Monkey Panel must be added to the current layout anywhere on the UI. There are some minor differences between the Default UI (DUI) and Columns UI (CUI), but in both cases the layout can be edited using the menu 'View\Layout\Live editing':

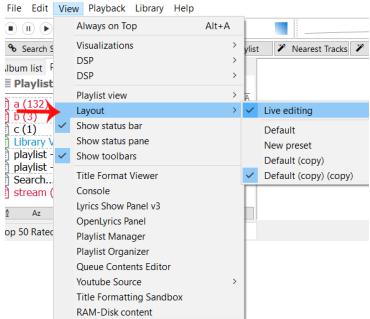


Figure 1: Editing the layout (CUI).

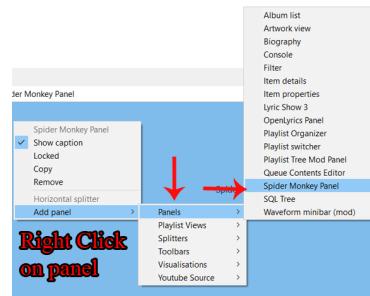


Figure 2: Adding a new SMP panel.

An SMP panel is only a blank UI panel with no further functionality until a '.js' script file is loaded. Multiple SMP panels can be added to a layout and they may point to the same script files without any problem; every panel is considered an independent entity. Multiple panels with Playlist Tools button are not supported (but the other tools are fine). After the panel has been added to the current UI, it must be configured to load the main '.js' script file. It can be done by right clicking on the blank SMP panel:

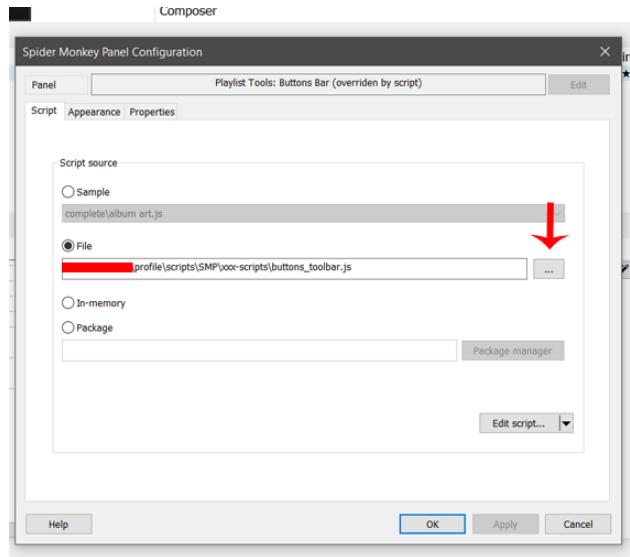


Figure 3: Loading the '**buttons\_toolbar.js**' script within a SMP panel.

Once the script has been installed, buttons may be added [12] and presets used [11].

## 5 Installation as a toolbar panel

-requires CUI-

This installation variant requires Columns UI to be used. Foobar2000 allows to display at top an indefinite number of toolbars (rows), and it's possible to integrate SMP panels into them using CUI.

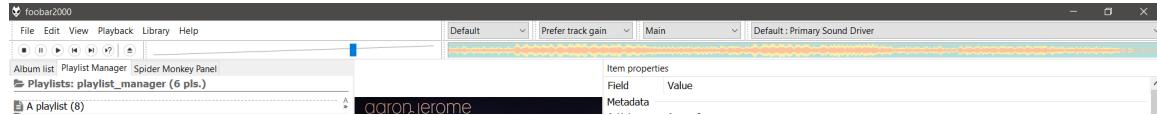


Figure 4: A pretty standard installation with 8 toolbar panels (see dotted lines) in 2 rows.

To do so, right click on the main menus toolbar (the blank space or even the menus), and click on 'Panels\Spider Monkey Panel'. It also works clicking on the dotted line at left of every row (to insert it on specific rows). Afterwards, follow the default installation steps [4] seen before, by right clicking on the new panel.

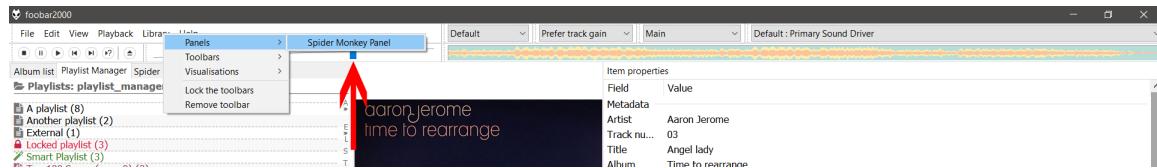


Figure 5: Toolbar configuration menu at right click.

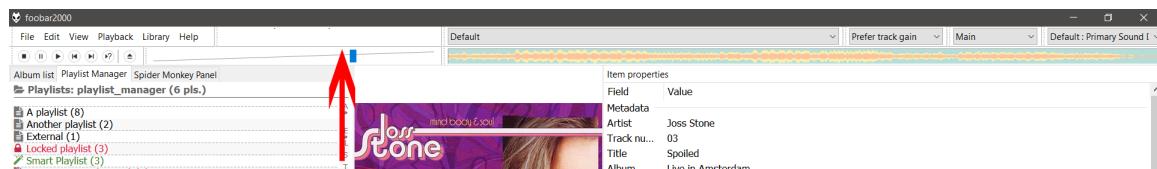


Figure 6: A new toolbar SMP panel will appear, with a white background.

Clicking again on the same menu entry will remove the panel but **there is a trick to create more panels: Press shift while clicking on 'Panels\Spider Monkey Panel'**. It will append new toolbar panels to the existing ones. Pressing without shift, when there are multiple panels created this way, will remove the first one available starting from the current row.

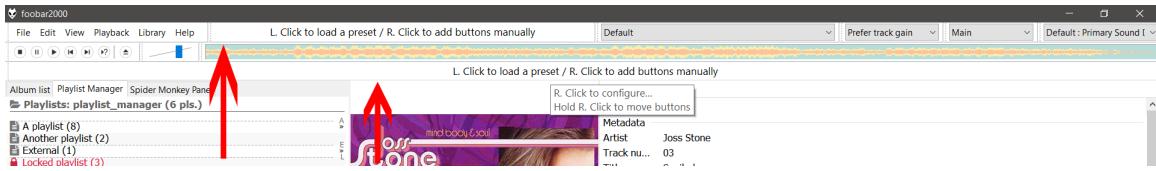


Figure 7: 2 SMP buttons toolbar panels already installed.

As any native toolbar, they may be moved, configured or resized clicking on the dotted line at their left.

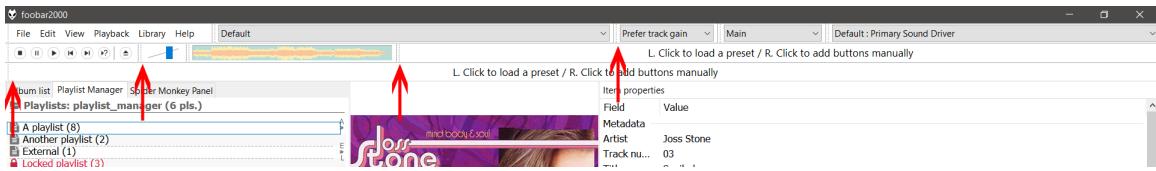


Figure 8:

It may be wise to set the edge style to 'None' and check 'Use pseudo-transparency' at the SMP configuration panel\Appearance, so the panel is integrated following the current global theme (instead of the white background seen below).

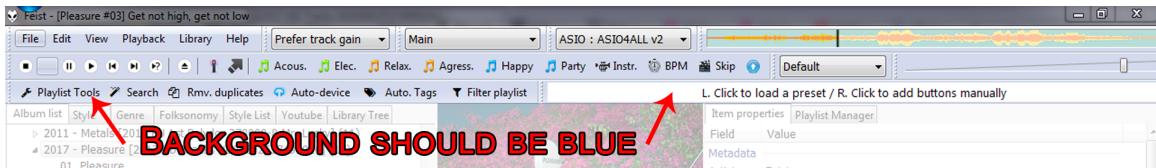


Figure 9: In case Windows uses a colored theme, the panel color mismatch will be obvious.

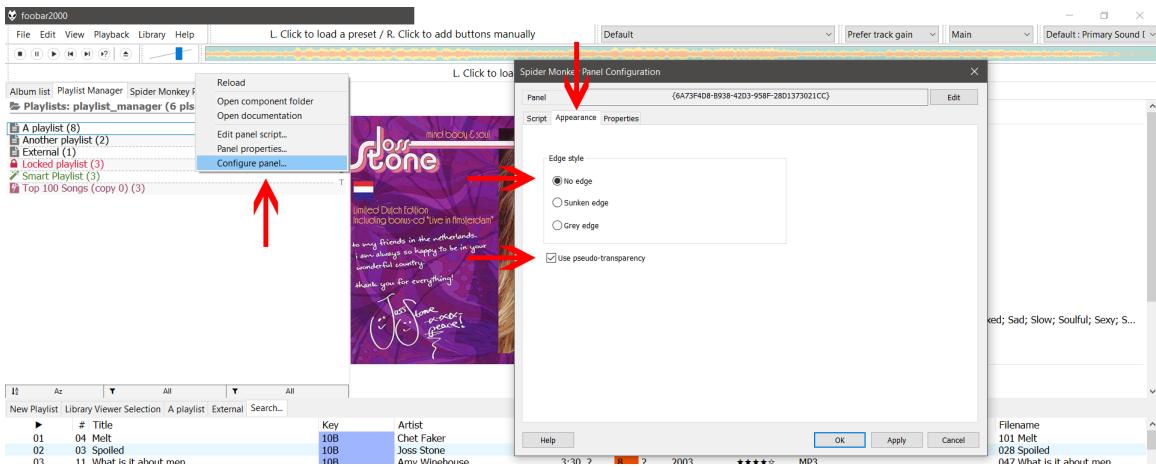


Figure 10: Configuring appearance.

## 5.1 Advantages over standard panels

Following all these steps, SMP buttons panels work exactly the same than native buttons. This installation variant has also multiple advantages:

- Theme is managed by native Foobar2000.
- Size is automatically adjusted and standardized between all toolbars.
- There is no need to rearrange the current layout for new panels, it works in any layout as an ad-hoc toolbar.
- Panel(s) can be easily moved or resized as the default toolbars without affecting the layout.
- It's trivial to create multiple toolbar panels in the same row. This could be done for performance reasons when using resource intensive-tools (since one button may be blocked "processing" but the other panels are independent).
- While the same results may be achieved using the layout installation method, it usually requires a lot more of work or thought to integrate it properly on complex themes.

**Note standard panels size can always be adjusted (so they can be reduced to fit the buttons size), while toolbar panels are fixed on height and only width can be changed.**

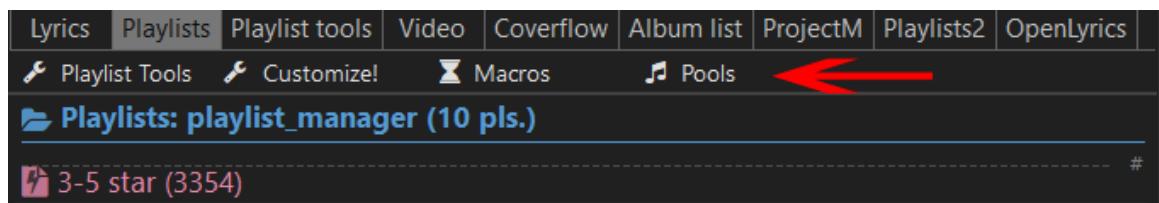


Figure 11: Standard panel, with 4 buttons, resized to fit buttons height.

## 6 Installation along Buttons framework compatible scripts

When using this script or other button tools, you may want to use the tools within an existing toolbar panel (instead of creating another independent panel). To do so skip loading the button file within a new panel and go to the toolbar section [III] for further instructions. A non comprehensive list of compatible button-scripts includes:

- Fingerprint-Tools-SMP
- Device-Priority-SMP
- ListenBrainz-SMP
- Search-by-Distance-SMP
- ...
- For the full list look [here](#).

## 7 Wine installations

Script is 100% compatible with unix systems although some points should be taken into consideration:

- 'Segoe UI' must be present, as it is the font by default on the script's UI.
- In case the font is not found (check console log), revise your default font on Wine. Changing it to a different one than 'Segoe UI' may give problems or crashes in some cases.
- [Additional] required fonts must be installed to avoid crashes. Is not only eyecandy.
- At some point the script may use external CMD tools which must be compatible with 32 bit systems in Wine (since that's the most popular version, with IE). In this case 32 bit binaries for 7z and CmdUtils are bundled.
- Script may throw an error after installation (either a foobar crash or panel crash) due to 'missing files'. There is a fix for that, check [IX] and [X]. It's a SMP bug, don't report it.
- Wine installations are specially sensible to missing files, wrong paths, etc. which may lead to crashes or bugs usually not found on Windows installations. Have that in mind and double check your installation procedure before reporting an error [X].
- All scripts try to use wine-friendly methods, focusing on configuration settings that can be changed via menus or the UI panel, instead of using HTML (which only works on Windows), known working fonts, etc.
- Please read SMP Wine page and feel free to report (me) any additional problem with these scripts.
- Read Wine foobar thread for more info and tips.

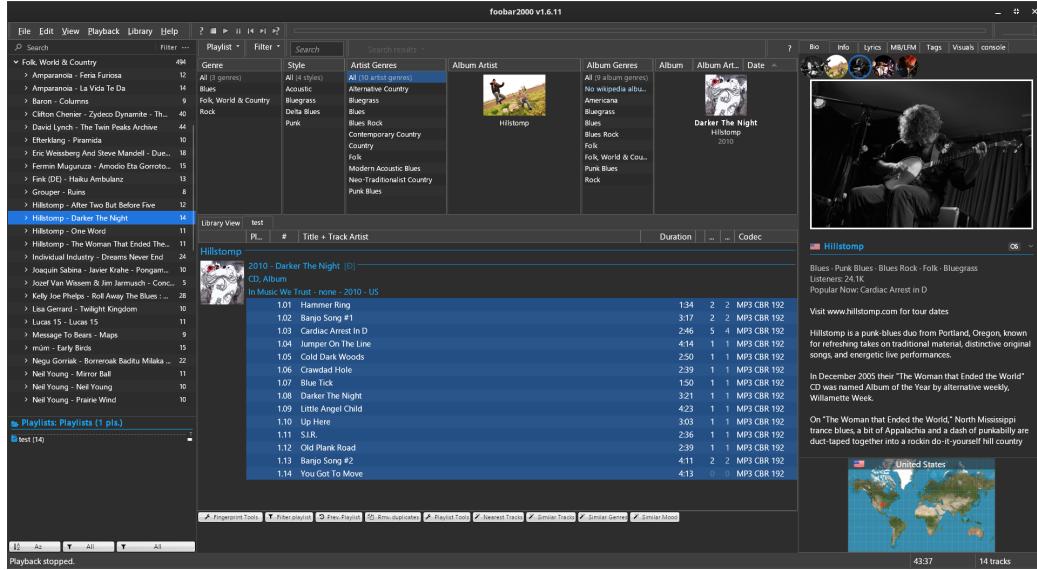


Figure 12: Script working on Wine along Playlist Tools and World Map.

## 8 Updating from older script versions

There is nothing special to do but following the standard installation steps and overwriting all [old] files as needed. In case an update changes something relevant or requires an additional step, that will be warned on the release itself.

## 9 Nightly versions

it's possible to download the more recent version directly from the repository, instead of the releases page.

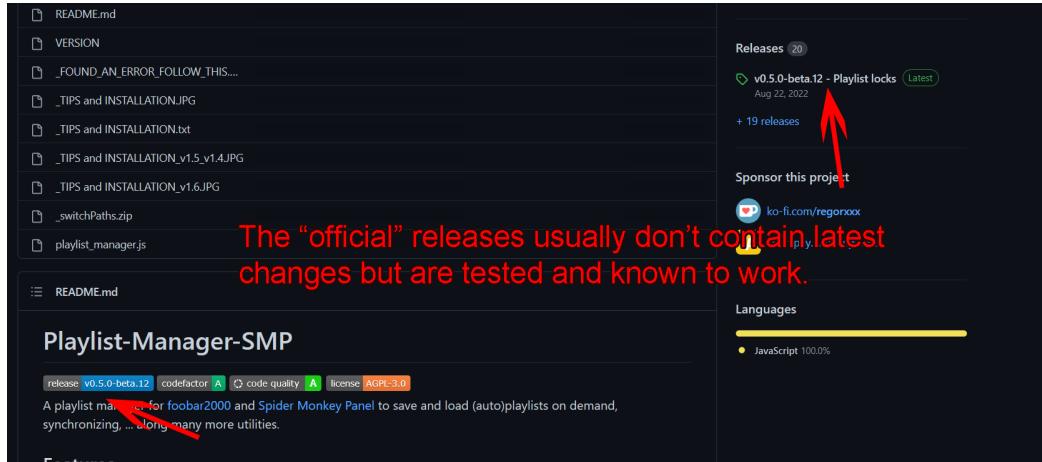


Figure 13: Releases vs WIP.

In such case it's possible that some changes are not fully tested; the list of "unreleased" changes is always shown -and updated- at the changelog. The installation instructions are the same than the other cases. To download the current repository, look below:

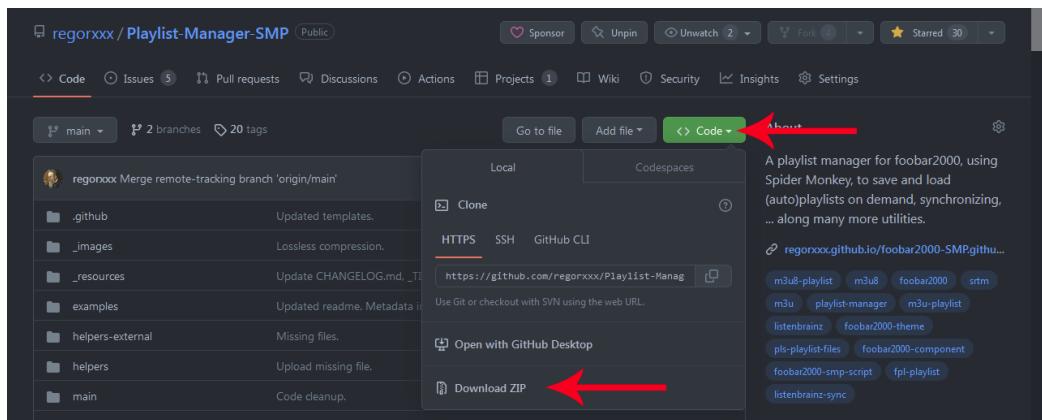


Figure 14: Downloading most recent files on repository as zip.

## Part III

# Buttons and Toolbar framework

## 10 Introduction

A quick lookup to most of my scripts would reveal many of them are meant to be used as buttons on a toolbar. In consequence, installing them as independent panels would easily clutter most layouts... to solve this inconvenience, all my button scripts have been built to be loaded on a single panel (if desired) over a single toolbar SMP script panel.

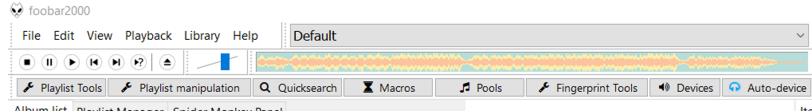


Figure 15: Multiple tools on the same panel/toolbar.

## 11 Loading preset

On first installation a popup should appear (or when clicking on the panel) asking to load a preset. Presets are just pre-defined layouts for the toolbar, where some tools are pre-configured.



Figure 16: A toolbar without buttons loaded.

Available presets are dependent on the scripts installed, i.e. if some of the tools listed here [6] are missing, then those presets are hidden. Nothing stops you to add/remove buttons at a later stage though.

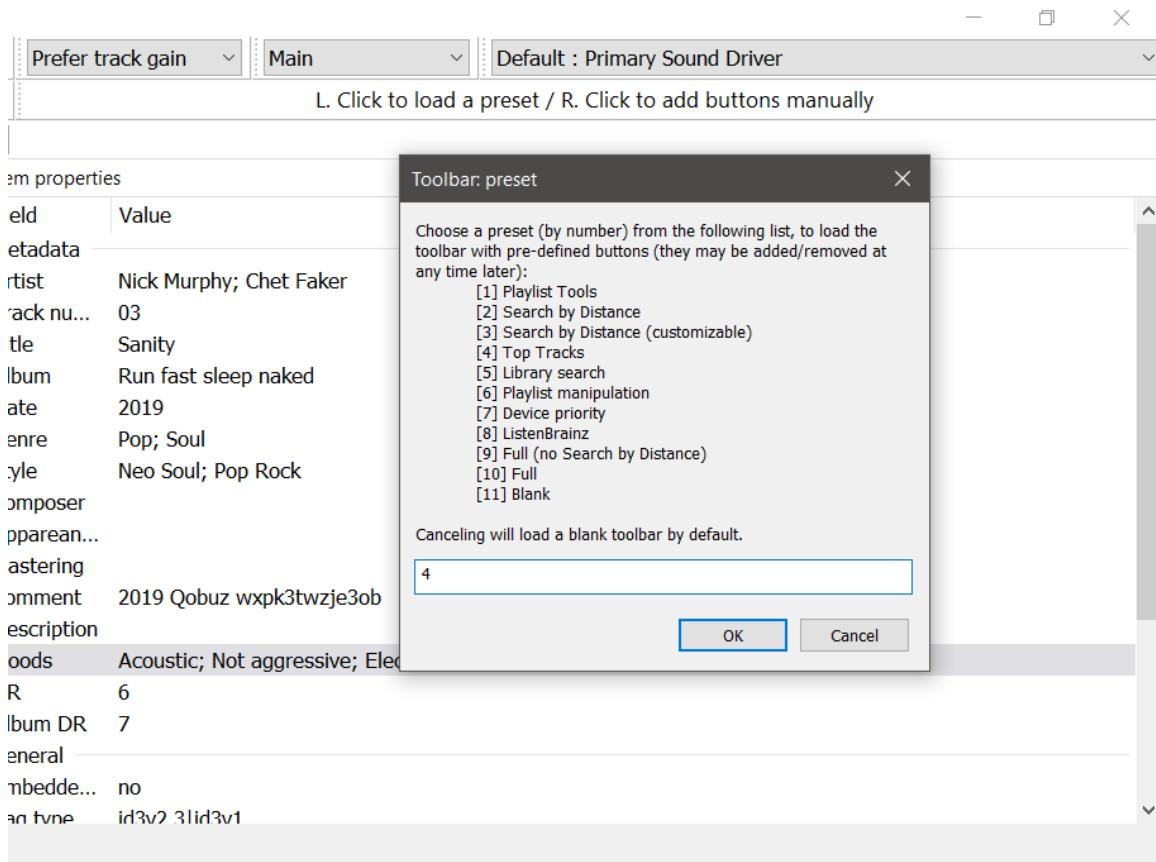


Figure 17: List of available presets.

## 12 Adding buttons

In case loading manually some buttons is desired, it doesn't matter whether the actual toolbar is using a preset or not. Right click -anywhere- on the toolbar and use the 'Add buttons' sub-menu. Buttons are categorized by functionality so they can be easily found.

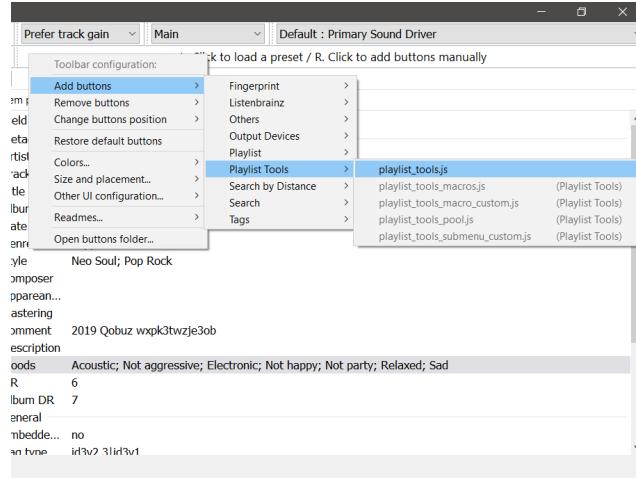


Figure 18: Sub-menu for adding buttons. Some of them require others added first.

Some buttons may allow only one copy per panel, in such case the entry will be greyed out if there is already one. Also when a pre-requisite button is not present yet (see above).

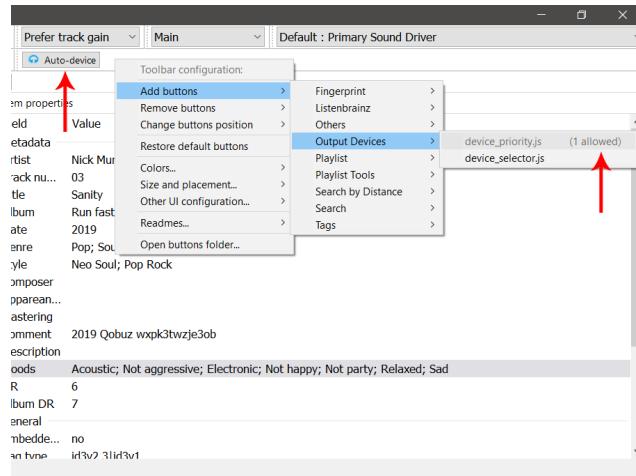


Figure 19: Only one copy of some button is allowed.

## 13 Removing buttons

Buttons may be easily removed using the appropriate sub-menu. The number displayed at right correspond to the position of the button (relevant when there are multiple copies).

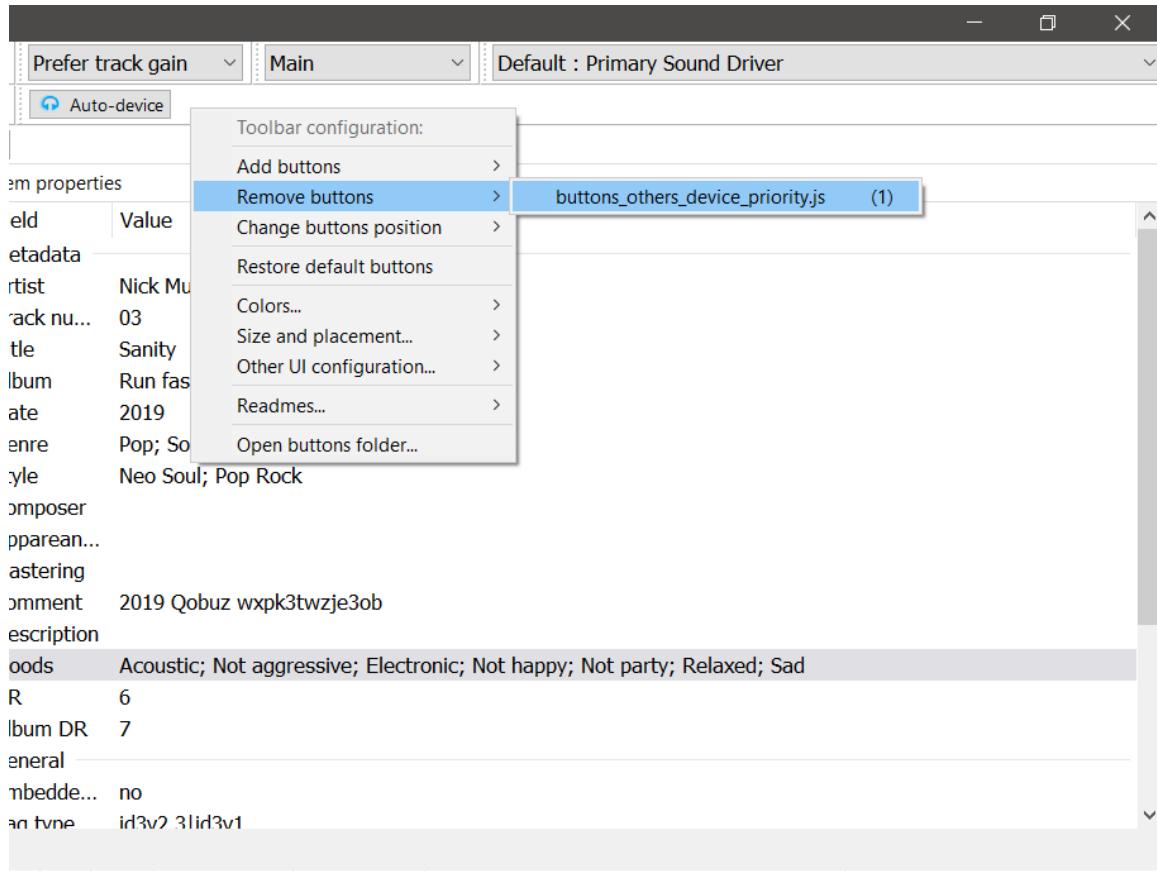


Figure 20: Sub-menu for removing buttons.

## 14 Changing buttons position

There are 2 ways to move buttons, menu or drag-n-drop. The former uses the 'Change buttons position' sub-menu, which will ask the user to enter the new position via popups. The selected button will be inserted at such position and the others shifted as necessary.

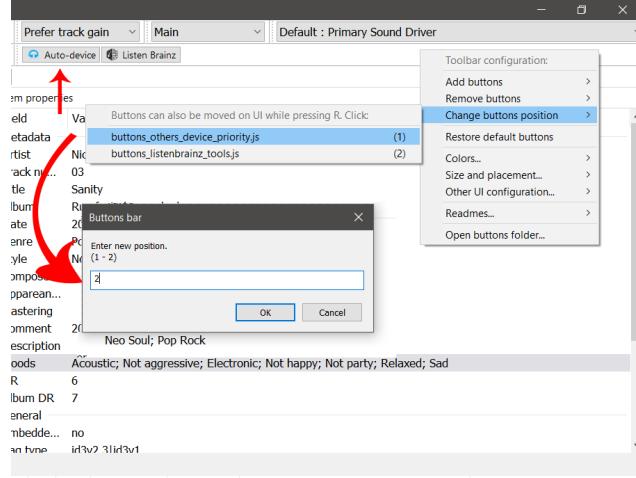


Figure 21: Moving buttons with menu.

The later allows to move buttons while pressing R. Click. A box will appear on the currently selected slot, and releasing the mouse will move the button to it.

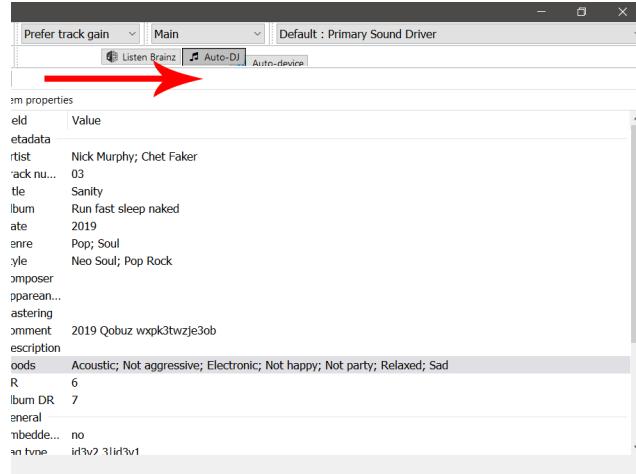


Figure 22: Drag-n-drop button to new position (black box).

## 15 Toolbar orientation

On the previous installation steps [4] multiple ways to add the buttons panel have been shown. They have one thing in common though, all buttons are displayed in a X axis.

It's possible to configure the panel to use Y axis and add buttons from top to bottom. In such case, a slim panel at the sides would be preferable:

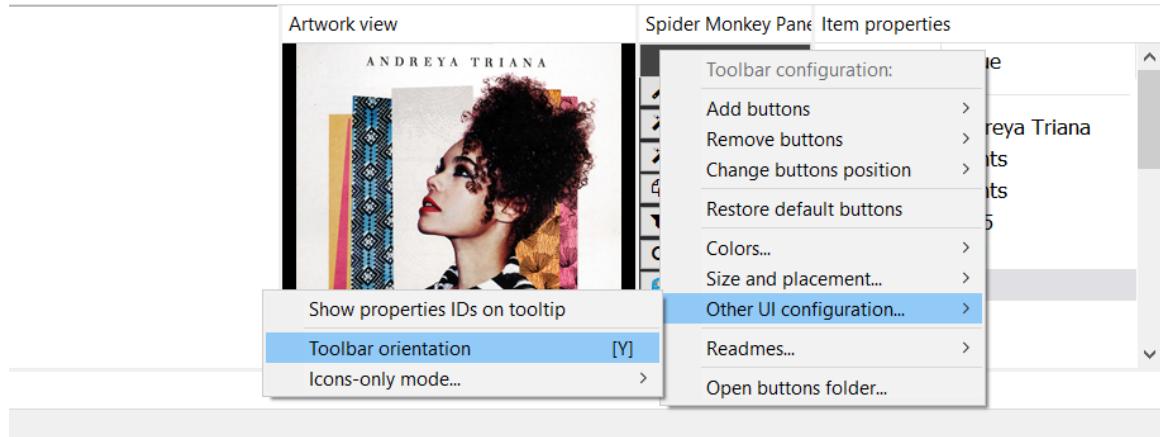


Figure 23: Setting panel to use Y axis.

The screenshot shows the Spider Monkey Panel in Y-axis orientation. On the left is the 'Artwork view' showing an album cover for 'ANDREYA TRIANA GIANTS'. To the right is a 'Spider Monkey Panel' with a grid of buttons. A red arrow points from the 'Toolbar orientation' setting in Figure 23 to the vertical stack of buttons in this panel. To the right of the buttons is a 'Item properties' table with columns for 'Field' and 'Value'. The table includes fields like 'Artist' (Andrey), 'Title' (Giants), 'Album' (Giants), 'Date' (2015), 'Genre' (Soul), 'Composer' (empty), 'Performer' (empty), and 'Album artist' (empty). At the bottom, there is a table of music files with columns for 'Date', 'Rating', 'Path', and 'Filename'. The table lists tracks from 2010 to 2015, each with a star rating and an MP3 file extension. The width of the buttons in the panel is normalized, making them appear smaller than in the X-axis orientation.

Date	Rating	Path	Filename
2015	★★★★☆	MP3	024 It's not over
2015	★★★★★	MP3	132 Giants
2010	★★★★☆	MP3	129 Daydreamer
2010	★★★★★	MP3	104 A town calle
2015	★★★★☆	MP3	105 The changing
2010	★★★★☆	MP3	017 Lost where I
2015	★★★★☆	MP3	075 Keen running

Figure 24: Buttons panel on Y axis. Note the buttons width has been normalized.

## 16 Reflow buttons

By default, the number of allowed buttons is limited by the width (or height in Y orientation). i.e. while adding new buttons works, they will be "hidden" since there is no available space. Buttons size is normalized by default to ensure all rows or columns have the same size.

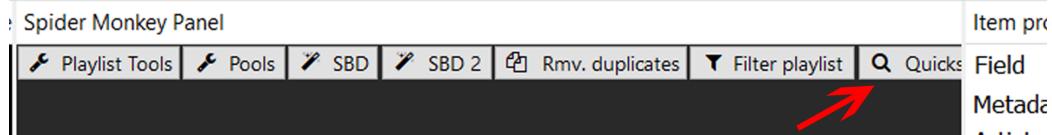


Figure 25: Not enough width to display all the buttons.

As an alternative, a setting may be enabled to distribute the buttons between multiple rows. Whenever a button would not fit on the current row, it would be pushed to the next one.

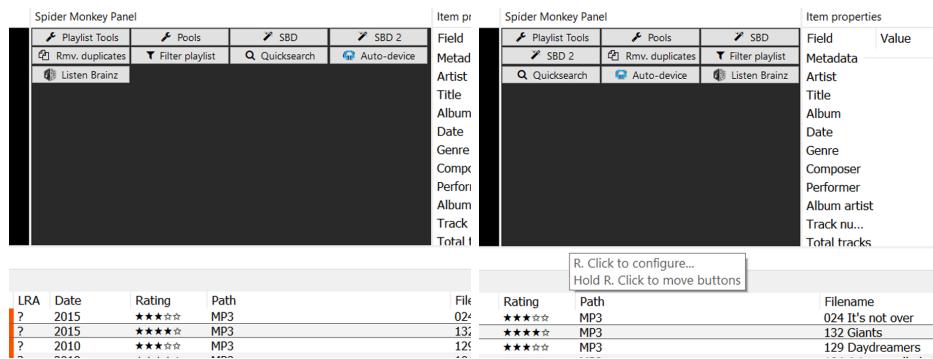


Figure 26: Panel reflow according to width on X orientation.

For Y orientations same logic applies, pushing buttons to next column whenever total height is greater than available panel height.

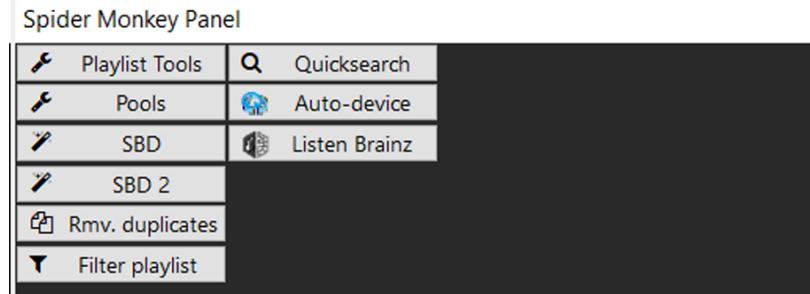


Figure 27: Reflow on Y orientation.

## 17 Icons-only mode

There is also an option to hide all button's text and display only the icons. They may be expanded -to show the name- on mouse hover.

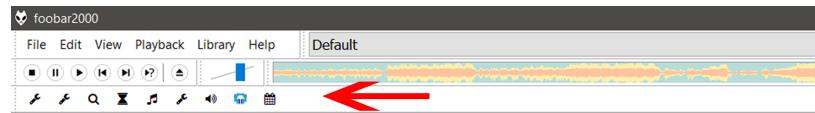


Figure 28: Toolbar panel on X axis using icons-only mode.

A screenshot of the foobar2000 interface showing the 'Artwork view' on the left, displaying an album cover for 'Andreya Triana - GIANTS'. On the right, there are two panels: 'Spider' and 'Item properties'. The 'Spider' panel is collapsed. The 'Item properties' panel is expanded, showing a table of metadata fields and their values. A red arrow points to the right side of the 'Item properties' panel, where the field names are visible as tool tips. Below these panels is a library view showing a list of tracks with columns for Item, Rating, Path, and Filename.

Figure 29: Buttons panel on Y axis using icons-only mode.

Whenever this mode is enabled, either globally or for individual buttons, the tooltip will also display the button's name to ensure all the info is available.

## **Part IV**

# **UI**

### **18 Features**

-WIP-

#### **18.1 Tooltip**

-WIP-

## **19 Customization**

### **19.1 Custom color**

-WIP-

## Part V

# Other scripts integration

This is a non comprehensive list of other Spider Monkey Scripts or plugins which may be used along the manager or whose features are designed to work together:

### - Playlist-Manager-SMP:

- \* **Random Pools:** Pools may use tracks from playlists files tracked by the manager, not requiring to have the playlists loaded within foobar. i.e. Random Pools component-like playlist creation, using not only queries as sources, but also other playlists or playlists files. See [22].
- \* **Playlist Revive:** Finds and replaces dead items on loaded playlists or selection. Meant to be used along dead items checks on playlist files. First check all playlist files, then load those with dead items and use Playlist Revive.
- \* **Duplicates and tag filtering:** The manager allows to report playlists with duplicated items, but it's limited to entries with same path. This tool expands Foobar2000 native functionality of removing duplicates, allowing to find duplicates by tags (for ex. any track with same Title - Artist).
- \* **Import track list:** Takes a plain text list of tracks (for ex. Title - Artist) and finds matches on library to create a playlist. Meant to be used for playlist importing when the track list does not follow an standard format or there are no paths provided<sup>1</sup>. Instead of sharing a list of files, list of tracks may be used which work universally no matter your configuration. Non found items are simply discarded.

---

<sup>1</sup>Technically that is not a playlist. But note playlists with relative paths may easily be considered a track list as long as you discard the '\.' part. In other words, a plain-text list can be retrieved from playlists in many cases.

## Part VI

# FAQ

- **What's json?** It's a standard file structure. Check <https://en.wikipedia.org/wiki/JSON> for more info.
- **What's asynchronous execution?** Execution of some code done on the background (usually on iterative steps), thus not blocking the UI on the process. For ex. external playlist files loading on native Foobar2000
- **Some lines in imported text files are not displayed/recognized properly:** That probably points to code-page detection errors by SMP on reading. UTF-8 files without BOM may be incorrectly identified and thus not parsed as intended, consider using UTF-8 with BOM before importing track lists. Multiple code checks have been added to minimize these situations in any case.
- **The tools don't work as expected at random instances:** Check the installation process and paths. Take a look at the file named '*\_TIPS and INSTALLATION.txt*' which should be found along this readme.
- **I have found a possible bug not related to a wrong installation:** please, create a new issue at github: <https://github.com/regorxxx/Playlist-Tools-SMP>.

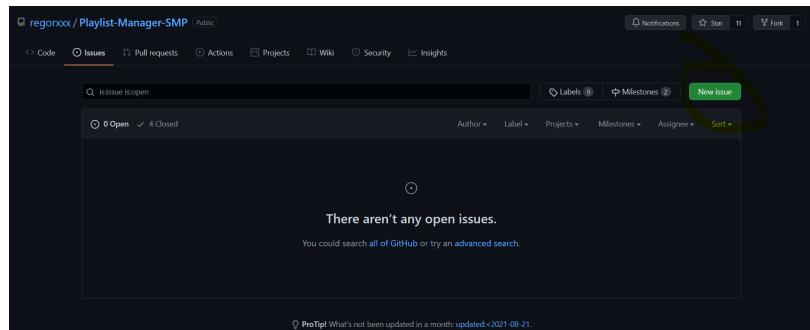


Figure 30: Opening a new issue at github.

## Part VII

# Advanced tips

## 20 Example

- -WIP-

## 21 Portable 'plug&play' installation

- **Real portable installations (i.e. on a external drive, network installations, etc.) may need to track some configuration files using their relative paths instead of absolute paths to work properly... otherwise they will not be found the tracked folder as soon as the drive letter changes.**
- **Relative paths for files always are checked against foobar path:** this is true only at places like the properties panel, etc. i.e. '.\profile\js\_data\' equals to 'D:\foobar2000\profile\js\_data\'.
- Once the panel is set properly, it just takes a matter of seconds to copy the entire panel 'as is' to other installations: the config files (FOOBAR\_PROFILE\_PATH\js\_data\\*.json) and properties panel (can be saved as json) can be transferred without changes.

## 22 Pools using Playlist-Manager-SMP

- Feature is enabled automatically when using both scripts in the same foobar2000 instance. As previously noted, Playlist-Tools-SMP [V] has a random pools feature which can use playlists as sources to output a random list of tracks from all its pools. Playlist-Tools-SMP checks periodically if there is a Playlist Manager panel and retrieves the list of playlist being tracked to use them as source if required. In other words: there is no difference between loaded playlists within the UI or playlist files.
- There is a preset example at "presets\Playlist Tools\pools\test\_playlistManager.json" which uses a playlist named "test" as source (after importing it). The file may be used as reference to create your own presets.

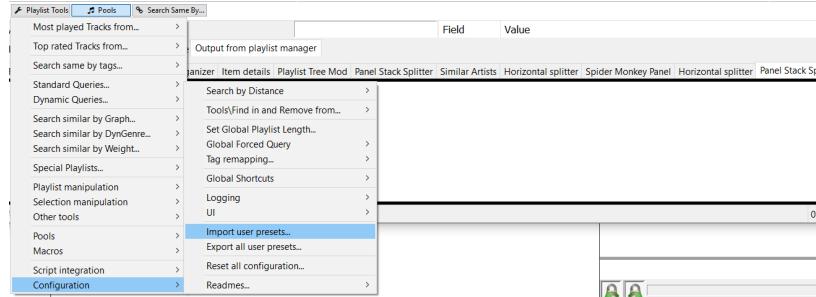


Figure 31: Importing a preset file in Playlist-Tools-SMP.

- For example to use "Playlist A" and "Playlist B" as sources, a preset file would have to be set like this:

```
{  
    "readme": "Playlist A and Playlist B as sources, 25 tracks per playlist.",  
    "pools": [{  
        "name": "Playlist Manager test V2",  
        "pool": {  
            "fromPls": {  
                "Playlist A": 25,  
                "Playlist B": 25  
            },  
            "query": {  
                "Playlist A": "ALL",  
                "Playlist B": "ALL"  
            },  
            "toPls": "Output playlist",  
            "sort": "%playlist_index%",  
            "pickMethod": {  
                "Playlist A": "random",  
                "Playlist B": "random"  
            }  
        }  
    }]  
}
```

- Alternatively, a **custom pool** may be directly executed once via menus -or added as a new entry for later use-:

If animation doesn't work, click to open the gif file on external viewer.

- **Playlists tracked by the manager(s) may be used along other sources** like playlists within the UI, library (+query) and other complex functions:

```
[...]
"fromPls": {
    "Playlist A": 25,
    "_LIBRARY_0": 7,
    "_SEARCHBYGRAPH_0": 13
},
```

- Playlist follow this rule when **trying to find a match**: loaded playlist first, then matching playlists by name (metadata) and finally by filename.

## **Part VIII**

# **Technical notes**

- -WIP-

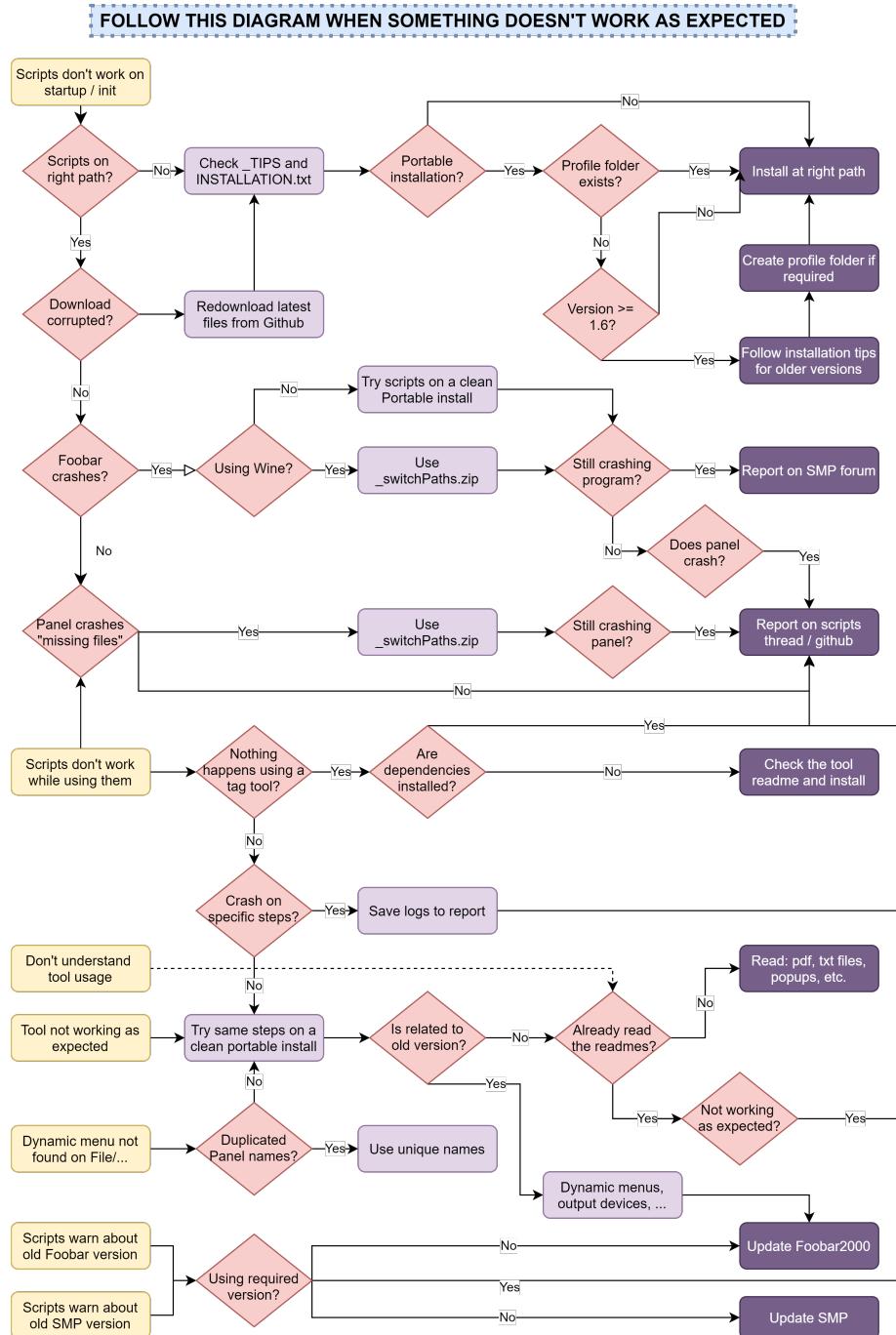
## Part IX

# Known problems

- Spider Monkey Panel 1.5.2 may crash -without known reason- at startup when trying to install the manager. The error will warn about script files not being found, although they are in place. This is due to incorrect relative path handling on some systems. In such case, take a look at the '`switchPaths.zip`', decompress it and follow its instructions. Scripts will be automatically edited to use absolute paths at script loading, thus fixing the problem. The cmd file (.bat) will have to be rerun on future playlist manager updates. Obviously this is only a workaround and a proper fix is expected on posterior SMP releases.
- Spider Monkey Panel (any version) may incorrectly identify code-page while reading text files, thus reporting wrong values at some instances. This usually happens while reading playlist files or importing tracks lists, with track names or artists using exotic chars (ä,å, ...) getting corrupted. Multiple code checks have been added to minimize these situations in any case. See FAQ [VI].

## Part X

# Bug procedure



## Part XI

# License

 regorxx/Playlist-Manager-SMP is licensed under the <b>GNU Affero General Public License v3.0</b>	<b>Permissions</b> ✓ Commercial use ✓ Modification ✓ Distribution ✓ Patent use ✓ Private use	<b>Limitations</b> ✗ Liability ✗ Warranty	<b>Conditions</b> ⓘ License and copyright notice ⓘ State changes ⓘ Disclose source ⓘ Network use is distribution ⓘ Same license
Permissions of this strongest copyleft license are conditioned on making available complete source code of licensed works and modifications, which include larger works using a licensed work, under the same license. Copyright and license notices must be preserved. Contributors provide an express grant of patent rights. When a modified version is used to provide a service over a network, the complete source code of the modified version must be made available.			
This is not legal advice. Learn more about repository licenses.			

Figure 32: License abstract.

### GNU AFFERO GENERAL PUBLIC LICENSE Version 3, 19 November 2007 Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

#### Preamble

The GNU Affero General Public License is a free, copyleft license for software and other kinds of works, specifically designed to ensure cooperation with the community in the case of network server software.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, our General Public Licenses are intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

Developers that use our General Public Licenses protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License which gives you legal permission to copy, distribute and/or modify the software.

A secondary benefit of defending all users' freedom is that improvements made in alternate versions of the program, if they receive widespread use, become available for other developers to incorporate. Many developers of free software are heartened and encouraged by the resulting cooperation. However, in the case of software used on network servers, this result may fail to come about. The GNU General Public License permits making a modified version and letting the public access it on a server without ever releasing its source code to the public.

The GNU Affero General Public License is designed specifically to ensure that, in such cases, the modified source code becomes available to the community. It requires the operator of a network server to provide the source code of the modified version running there to the users of that server. Therefore, public use of a modified version, on a publicly accessible server, gives the public access to the source code of the modified version.

An older license, called the Affero General Public License and published by Affero, was designed to accomplish similar goals. This is different license, not a version of the Affero GPL, but Affero has released a new version of the Affero GPL which permits relicensing under this license.

The precise terms and conditions for copying, distribution and modification follow.

## TERMS AND CONDITIONS

### 0. Definitions.

"This License" refers to version 3 of the GNU Affero General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

### 1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming

language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

## 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

## 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or

modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

#### 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

#### 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

#### 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

## 7. Additional Terms.

”Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered ”further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying

under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

#### 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

#### 9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

#### 10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

## 11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products

or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

#### 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

#### 13. Remote Network Interaction; Use with the GNU General Public License.

Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a computer network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software. This Corresponding Source shall include the Corresponding Source for any work covered by version 3 of the GNU General Public License that is incorporated pursuant to the following paragraph.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the work with which it is combined will remain governed by version 3 of the GNU General Public License.

#### 14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU Affero General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU Affero General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU Affero General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU Affero General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS