



# TSWTools

Toolkit for TrainSimWorld

Users guide

Rudolf Heijink

Version 0.5 alpha

Copyright © 2018/2019 Rudolf Heijink.



*This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.*

### **You are free to:**

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material

The licensor cannot revoke these freedoms as long as you follow the license terms.

### **Under the following terms:**



**Attribution** — You must give [appropriate credit](#), provide a link to the license, and [indicate if changes were made](#). You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.



**NonCommercial** — You may not use the material for [commercial purposes](#).



**ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the [same license](#) as the original.

**No additional restrictions** — You may not apply legal terms or [technological measures](#) that legally restrict others from doing anything the license permits.

### **Notices:**

- You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable [exception or limitation](#).
- No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as [publicity, privacy, or moral rights](#) may limit how you use the material.

## Preface

### ***Introduction***

The last years I created a number of manuals and tools for DTG TrainSimulator. Mike Simpson, the author of the world famous RWTools has been an important source of inspiration to me. I never tried to copy his work, but I found some niches specifically for scenario authors that are not covered by RWTools. Mike announced he will not create such a toolkit for TSW. So I decided I could claim the TSWTools name with a lot of respect and thankfulness to Mike. I cannot but admire his perseverance in reverse engineering undocumented features without any support from DTG.

Here it is, the third alpha edition of TSWTools. It's not doing much yet, but the start is there.

### ***Acknowledgements***

All anonymous members of the TSW community for sharing their experience and helpfulness.

The creators of [Inno setup](#) for providing a free installer.

### ***License agreement***

The software and this guide are licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. Details are provided in the license agreement you must sign before using the software.

### ***Disclaimer***

This guide is provided "as is". The author is not liable for the consequences of the use of this guide or the LuaCreator application. The contents is the sole responsibility of the author.

### ***Contact***

Comments are welcome at [trainsimulator@hollandhiking.nl](mailto:trainsimulator@hollandhiking.nl).

But please be aware that I cannot provide you help with you scenario creating problems. If you have any questions, please use one of the regular community forums.

### ***Front page image***

One of my own screenshots made at the Sandpatch route. It still has a "wow" effect on me seeing all the details.

Rudolf Heijink

# Contents

<b>Preface .....</b>	<b>3</b>
<b>1 Introduction .....</b>	<b>7</b>
1.1 Next version.....	7
1.2 New in this version .....	8
1.2.1 Version 0.4.....	8
1.2.2 Version 0.3.....	8
1.2.3 Version 0.21.....	8
1.2.4 Version 0.2.....	9
1.2.5 Version 0.1.....	9
<b>2 Installation .....</b>	<b>10</b>
2.1 Installation procedure .....	10
2.2 Folder structure .....	10
2.3 Database.....	11
<b>3 User interface principles .....</b>	<b>12</b>
3.1 Modal versus modeless windows.....	12
3.2 Impact of the window size.....	13
3.3 Controls .....	14
<b>4 Main screen.....</b>	<b>18</b>
<b>5 Utilities.....</b>	<b>20</b>
5.1 Options Dialog .....	20
5.2 Log viewer.....	22
5.3 Backup save area .....	22
5.4 View key bindings .....	24
<b>6 Unpack and view .....</b>	<b>25</b>
6.1 Unpack game files.....	25
6.2 View unpacked files.....	27
6.3 Launch UModel .....	27
<b>7 Tools .....</b>	<b>29</b>
7.1 Game launcher .....	29
7.2 Railway Radio Stations .....	31
7.3 Edit TSW Settings.....	32
7.3.1 Introduction.....	32
7.3.2 Screen settings.....	33
7.3.3 Quality settings.....	34
7.3.4 Sound settings .....	35
7.3.5 HUD settings .....	35
7.3.6 Game play settings .....	36
7.3.7 Advanced settings.....	36
7.4 View screenshots.....	37
<b>8 Working with liveries and mods .....</b>	<b>39</b>
8.1 Introduction.....	39
8.2 Pak installer .....	40

8.2.1	Introduction.....	40
8.2.2	Pak installer details.....	41
8.3	Livery manager .....	41
8.3.1	Properties tab .....	42
8.3.2	Install tab .....	43
8.3.3	Sets tab .....	44
8.3.4	Livery manager glued .....	45
<b>9</b>	<b>Help.....</b>	<b>46</b>
9.1	About dialog .....	46
9.2	Open TSWTools manual .....	47
9.3	Open TSW Starters Guide.....	47
9.4	Open route guides .....	47
<b>A.</b>	<b>Download locations .....</b>	<b>48</b>
<b>B.</b>	<b>UModel command reference .....</b>	<b>49</b>
<b>C.</b>	<b>Known issues .....</b>	<b>51</b>
	<b>Index .....</b>	<b>52</b>

## List of figures

Figure 1 Folders used in TSWTools.....	11
Figure 2 Modeless windows .....	13
Figure 3 Example of a window with visible scroll bars .....	14
Figure 4 Buttons .....	14
Figure 5 Text Box .....	15
Figure 6 Result Text Box .....	15
Figure 7 Combo box.....	15
Figure 8 File Dialog .....	15
Figure 9 Radio buttons .....	16
Figure 10 Two coupled List Views .....	16
Figure 11 Slider control .....	17
Figure 12 Tab control.....	17
Figure 13 TSW Tools Main Screen .....	19
Figure 14 Options dialog, File Locations tab. ....	20
Figure 15 Options, Tab Other visible .....	21
Figure 16 Log viewer.....	22
Figure 17 Backup window .....	23
Figure 18 Key bindings or input mappings .....	24
Figure 19 Unpack tool window.....	26
Figure 20 Unpacker while working.....	27
Figure 21 UModel interface for viewing the contents of uasset files .....	28
Figure 22 TSW Launcher.....	30
Figure 23 Railway radio stations window.....	31
Figure 24 Settings editor .....	32
Figure 25 Screen settings.....	33
Figure 26 Quality settings.....	34
Figure 27 Sound settings .....	35
Figure 28 HUD settings .....	35
Figure 29 Game play settings .....	36
Figure 30 Advanced settings .....	36
Figure 31 Screenshot viewer .....	37
Figure 32 Screenshot preview window .....	38
Figure 33 Pak installer window.....	40
Figure 34 Livery manager Properties tab .....	42
Figure 35 Livery Manager Install tab .....	43
Figure 36 Livery Manager Sets tab .....	44
Figure 37 Livery management architecture .....	45
Figure 38 About Dialog .....	46
Figure 39 Route guides selector .....	47





## 1 Introduction

There is not yet much you can do to enhance your experiences with TrainSim World, but few things are useful and justify a tool. TSWTools will do this for you:

- Unpack all .pak files and make a local unpacked copy.
- View uasset file contents using the UModel application (experimental)
- View the key bindingsfile, used in the game
- You can edit most game settings in the tool, including enhancements like Viewdistance and extended sound levels. You make sets and save them outside the game
- Make backups of your user settings and progress and restore them
- Manage you additional liveries and add them to the game or remove them again
- A screenshot manager, that combines steam and TSW screenshots for easy selection.
- A game launcher, using a settings file, a livery set and may start a radio station url.
- Browse through the file locations

### 1.1 Next version

I will mainly work on a better performance of the screenshot tool. Also I would like to provide a solution to solve the issues you may experience on a lower resolution screen.

Maybe, if the content editing tools are available, I may provide some support functions, but nothing can be promised.

Error logging will be improved as well.

## 1.2 New in this version

### 1.2.1 Version 0.5

Improved functions:

- Added window sizing function, useful for smaller screens

Bugs fixed:

- Settings tool does not load Engine.ini from a save set due to error in file name
- Fixed issue that prevents opening route guides

### 1.2.2 Version 0.4

New functions:

- You now can define sets of liveries that will be included in the launch function.
- A new function to define URLs for Railway Radio stations, you can start one of the stations in the Launcher.
- A new setting, that helps to solve the bleach colouring of Sandpatch route.

Improved functions:

- The screenshot manager is improved
- The screen layout for the livery manager is now more compact
- Results textbox completely rehauled.

Bugs fixed:

- An issue with the launcher is solved now, so it does not end up in a warning about an additional parameter
- Objective Marker is now a Boolean toggle in the settings.
- Force feedback setting save issue solved.

### 1.2.3 Version 0.3

New functions:

1. A complete redesign of the user interface
2. Make a backup of your local game files (settings, progress, last played, screenshots ....)
3. A livery manager. You can store your additional liveries in a library and install them from this library.
4. A tool to edit most game settings, including some Unreal settings recommended by community members
5. Improved unpack tool
6. Improved game launcher
7. An interface to the UModel application, which you can use to view UASSET files
8. Error log screen

### 1.2.4 Version 0.21

- Fixed an annoying bug in the options menu that prevents to select the steam and TSW installation directory.



- Unpacking all TSW .pak files now processes all paks in parallel and shows a “busy” indicator while it is working.
- You now can unpack a single TSW game file (.pak type).

### **1.2.5 Version 0.2**

- Unpacking game files also works for DLC .pak files (all files will be unpacked)
- The screenshot viewer is improved, it will now handle your user id properly in most cases
- You can now save and restore several settings, e.g.
- A game launcher, which will load your options set and start Sandpatch Radio if you want that.

### **1.2.6 Version 0.1**

This is the first version. It does some basic stuff:

- You can easily unpack the game .pak files.
- You can view the files at the game location as well as the unpacked files
- There is a simple screenshot manager
- You can view the input mappings (but not edit them)
- You can view and edit the game settings outside the game.



## 2 Installation

### 2.1 Installation procedure

The game comes with an installer. Installing it is straight forward. The first time you run TSWTools, you must set some options in the game to enjoy all functions. See section 5.1

It has been tested using Windows 10, but likely will work in other environments as well. TSW Tools is written in C# and requires the latest .NET version to run.

You need to install the Unreal game engine and the UModel tool for some functions.

In order to read the manual, you need a pdf reader (by reading this it is clear to me you have that).

It is recommended to install a good text editor as well. Notepad++ is a good solution.

If you want to use the pak installer, you will need 7Zip installed.

Check out [www.hollandhiking/trainsimulator](http://www.hollandhiking/trainsimulator) for download links.

### 2.2 Folder structure

At the next page you can see the folder structure for the data TSWTools may store. The folders are created automatically after your first login. In rare cases this does not work, you can create them manually.

**Backup** is used to store saved backup sets.

**Liveries** contains the liveries. You are free to make a further subdivision. TSWTools will scan all subfolders to locate content.

**Manuals** is the place to go for documentation. In the root folder, you should find this manual, the starters guide and licence information. Inside the **RouteGuides** folder you can add subfolders as much as you see fit.

**OptionsSets** will contain the Options collections as you create them.

**Temp** is for temporary files. You can delete the contents if you need to.

**Unpack** contains the unpacked .pak files, each of them in its own folder structure. In the folder **UnpackedAssets** the UModel application should store its exports.

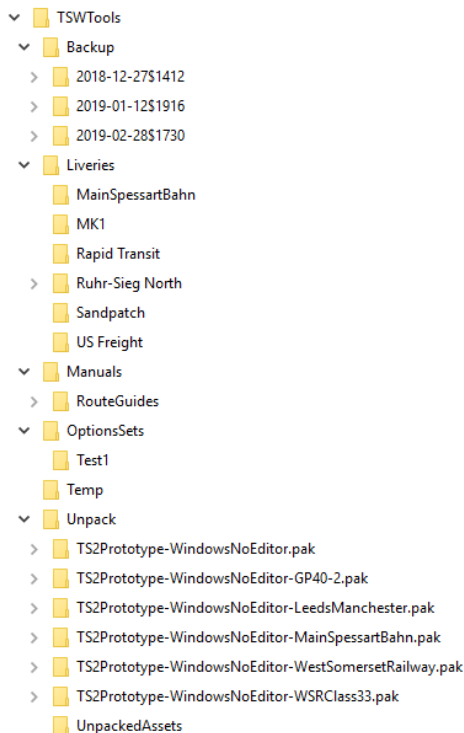


Figure 1 Folders used in TSWTools

## 2.3 Database

TSWTools uses a database to preserve some information for you. You can find it in the TSWTools root folder and it is called TSWTool.db. If it is missing, TSWTools will create it again for you, but all data will be lost.

If you want, you can open the database and see for yourself what is inside it, use the freeware tool you can download here:

<https://sqlitebrowser.org/>



### 3 User interface principles

In this chapter a short “buttons training”. This may help you to understand how the new user interface is designed and how you can use it. LuaCreator is a fairly complex application. It is not 100% fool proof, so handle it with a bit of care. The user interface also has more or less hidden features, you may consider convenient.

**Note:** this chapter is copied from another tool I created, called LuaCreator, which uses the same principles for its user interface. I did not yet update all examples specifically for TSWTools.

#### 3.1 Modal versus modeless windows

For Windows application, there are two ways to open a new window: **modal** or **modeless**.

A **modal window**, also called dialog form, needs to be closed using either the **OK button**, which usually saves data or a **Cancel button**, which cancels all changes made in the form. Access to all previously opened windows is blocked, while you are working with this window/dialog.

A **modeless window** spawns from its parent window. You can navigate freely between the windows you have open.

The advantage of modeless windows is a much larger flexibility and freedom during use. Freedom has its price, your desktop may soon be cluttered with a large number of open windows, and it is up to you to keep track of them. If data between these windows is related, changing data in a modeless window may cause inconsistency in other windows. Fortunately, WPF has some useful technologies to update all relevant windows automatically.

In the new WPF version of LuaCreator I decided to switch from Modal windows to modeless windows, with a very small number of exceptions. The main reason to do so, is that you always have access to the information you need. This works, because the logic in LuaCreator is (almost) completely separated from the logic to show data on the screen.

However, you can create situations where inconsistencies in data may occur. In this case, close the application and restart it.

You can close each window separately (using Alt+F4 or the cancel button in the top right position), all spawned windows will stay open. It is not a very good idea to close the startup window, at the moment there is no method to reopen it again, without closing TSWTools.

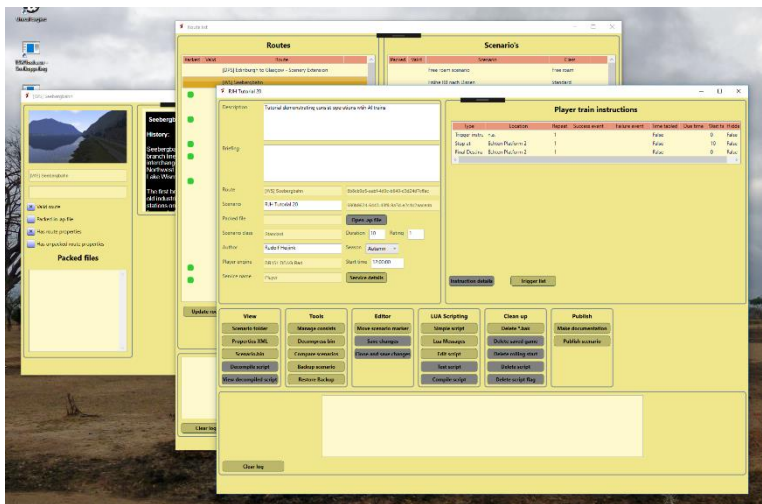


Figure 2 Modeless windows

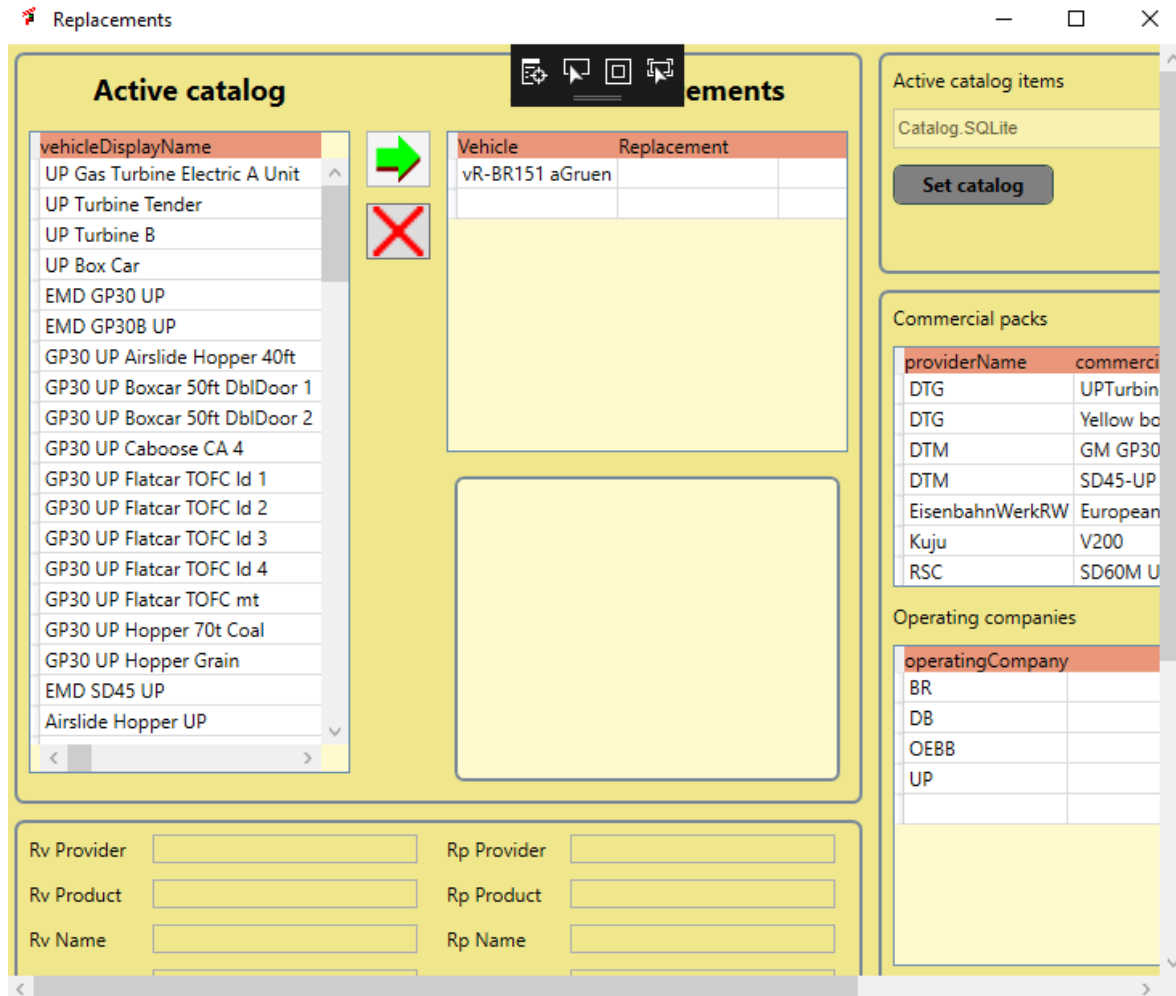
## 3.2 Impact of the window size

I created the new WPF version using a fairly large screen, with a resolution of 2450x1440 pixels. As I understood it, WPF would scale nicely for other screen sizes. Unfortunately this is not true.

So, when I tested TSWTools at my laptop screen (1920x1280 screen), the windows did not fit on the screen, which is not workable. For the short term I adapted some screens to fit better and adapt the content size a bit. I also added scroll bars to each window. Therefore if you work at a smaller screen it should be possible to use LuaCreator by scrolling. A consequence of all this is that full screen mode maximizes now to 09% of full screen.

For the next version, I will improve this a lot. This may take time, because for these scaling issues no tutorials exists and it seems little knowledge is documented. So I will need to find out how to do this exactly. Don't worry, I have some ideas.

**Note:** When working on a smaller screen e.g. a laptop screen, you may need to use the scroll bars to be able to use larger windows.



**Figure 3** Example of a window with visible scroll bars

### 3.3 Controls

TSWTools uses a large number of controls. WPF makes it relatively easy to give them similar look and feel. In LuaCreator simple styling is used, later these may be replaced by more advanced and fancy options. For now, it is mainly functional, so it is helpful to understand the concepts that are used.



**Figure 4** Buttons

**Buttons** are used to start an action. TSWTools uses colour codes to tell you more on the significance of a button:

- **Pink** is used for a **Cancel button**. This is mainly useful for a modal window and results in abandoning the planned changes.



- **Green** is used for the **OK button**, which usually makes changes permanent. For a modal window, an OK Button will close the window as well.
- **Grey** is used for a **Disabled button**. Pressing it has no effect, mostly because condition is not met. For instance, You cannot show Scenario Properties if no scenario is selected.
- **Dark Khaki** is used for **Normal buttons**. They will execute the indicated action.

**Figure 5 Text Box**

In Figure 5 you see a four elements. The left column contains the meaning of the right column (in this case Route and Scenario). The right column contains the actual value from the game data. You see the **Border line** for the text. If the background is **yellow**, this means you cannot change the text, if the background is **white**, you can edit the text in this field.

**Figure 7 Combo box**

A variant of this principle is a **combo box**, which shows a predefined set of values, e.g. the seasons in this example. The background is white, so you can edit it.

**Figure 6 Result Text Box**

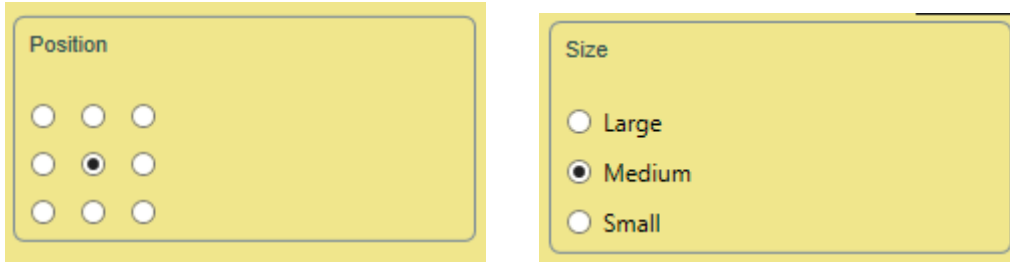
A special case is the **Result Text Box**. This Text Box has more than one line and it will automatically scroll to the last line when its contents changes. It now has an integrated button at the upper right corner to empty the contents, see the highlighted spot in Figure 6.

It is used to inform you about results of actions where needed. It's use is not always consistent yet, This will be repaired in next versions. On most windows you will find one.

A **File Dialog** also is a special case. It is used to select a **file** or **directory**. The **actual value** is shown in the Text Box, but as you can see you cannot directly edit it. If you want to change it, press the button at the right side showing three dots. This will open the well-known standard windows dialogs for selecting files or directories.

**Figure 8 File Dialog**





**Figure 9 Radio buttons**

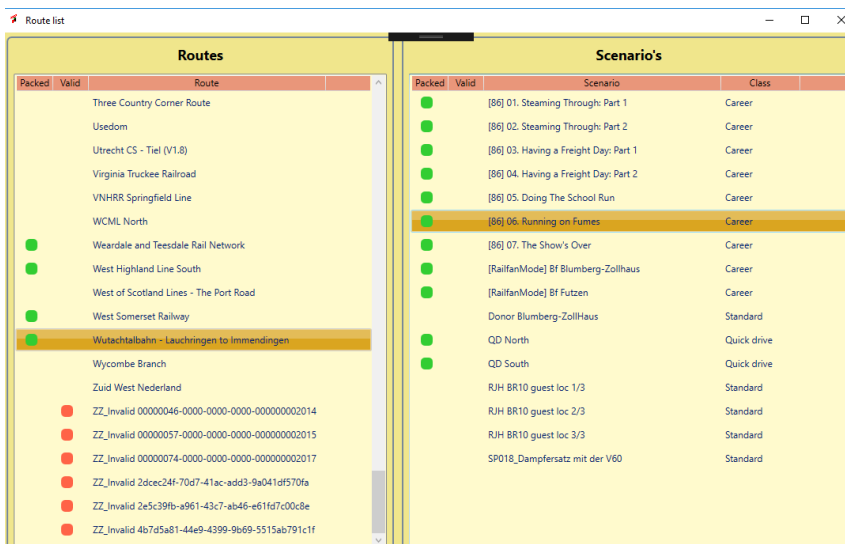
**Radio buttons** are sets of small round buttons you can check, but you can check only one out of a set. A set is surrounded by a thin border line. In one case, you see there is not any text. In this case each button represents a position at the screen for a Lua message. This will be explained later.

**Tooltips** are short help texts. You may see them at some screens. I intend to add them to all windows. Unfortunately, it is not possible to capture them easily.

TSWTools works a lot with **tables**. I have used two different technologies for tables:

1. **List View**
2. **Data Grid**

The bad news is that you cannot see which one I used. Data Grids offer a bit more flexibility than List Views. In the example below, you see the **Route List Window** which has two **List Views**. For newer windows I always will use a DataGrid.



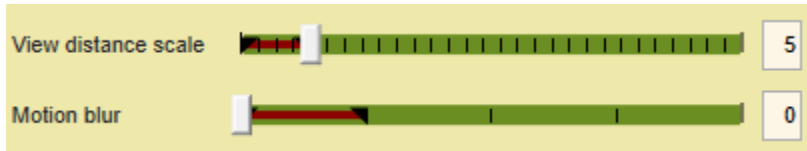
**Figure 10 Two coupled List Views**

Remarkable features (all of these are also valid for a Data Grid):

- The first line with the **Pink** background contains the explanatory **column headers**. You can make the columns wider or smaller if you need.
- A **selected line** is highlighted by giving it an **Orange** background colour.
- If the **number of rows** exceeds the space allowed for the table, **scroll bars** will be shown automatically.
- The **green** and **red blobs** act as a **check mark**. In this case a green blob means the route is packed in .ap file The red blob means the route is not a valid route.

A **Data Grid** has one additional feature:

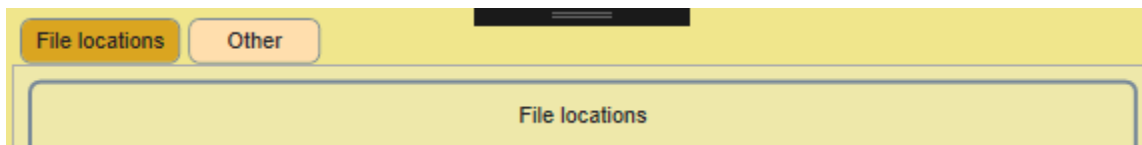
By **clicking** a column at a header row you can sort the table using this column.



**Figure 11** Slider control

The **Slider control** is used to select a value in a range. The brown area is a cosmetic feature and represents a recommended range. Please make sure to consult the documentation on the interpretation of this range. The tick marks show the granularity of the settings. In most cases the selectable values are restricted by the software. At the right side, you seen an TextBox. You also can type the value directly in the TextBox, but this is not the recommended practice.

As a last control, I will introduce the tab control. This looks a bit like a button. What it does is that you can hide parts of a window and show only one of them. For instance, the options window shows either all file locations or the other options. The main advantage is that it saves screen space and makes the window more compact.



**Figure 12** Tab control



## 4 Main screen

In Figure 13 the TSWTools main screen is depicted. Each function will be introduced in the subsequent sections.

1. Options, tool options. You need to set the options before you can use TSWTools
2. For errors, TSWTools provides an error log, which you can open here if needed.
3. Backup tool for the saved user data, screenshots etcetera
4. View the key bindings in a neat table format
5. Unpack the game .pak files using the unreal unpacker (takes a lot of time!)
6. View the unpacked files using explorer
7. Interface to the UModel toolkit, to view uasset files
8. Game launcher
9. Edit the game options and save options as a set
10. Manage your additional mods and liveries
11. Manage a list of Railway Radio Stations
12. The screenshot manager
13. An installer tool to install liveries directly from an archive.
14. About this tool
15. Open the TSWTools manual
16. Open the TSW Starters Guide
17. Open route guides

18. Close button, terminates TSWTools

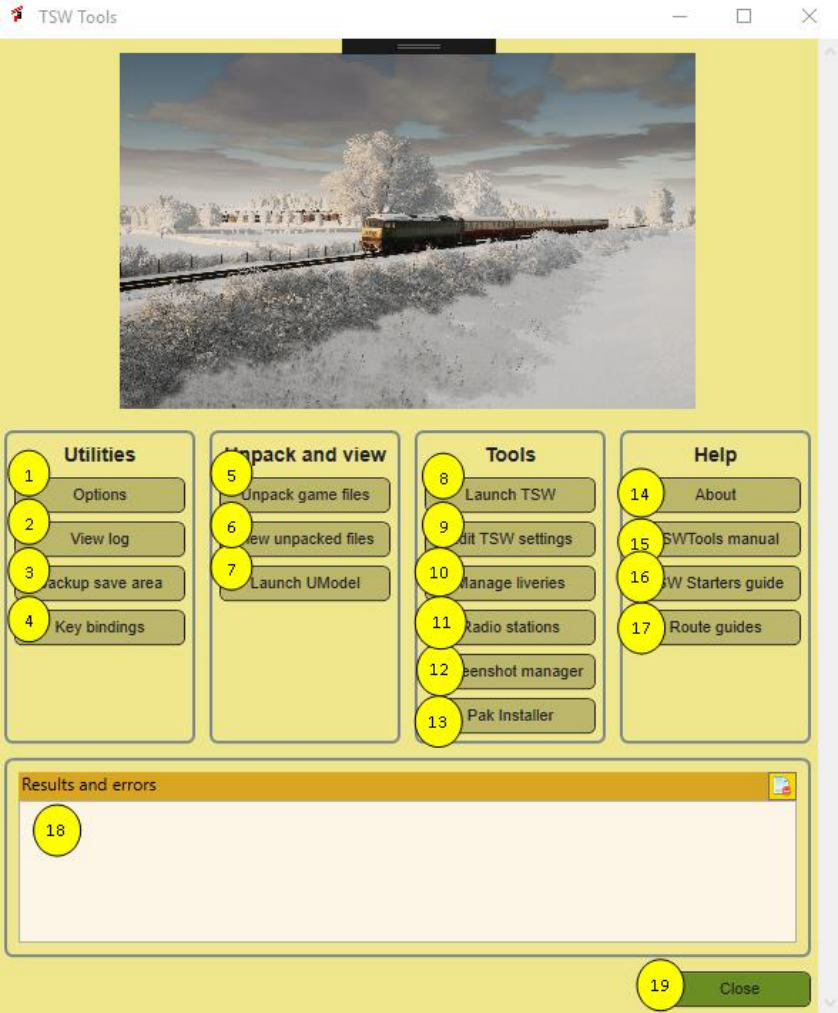


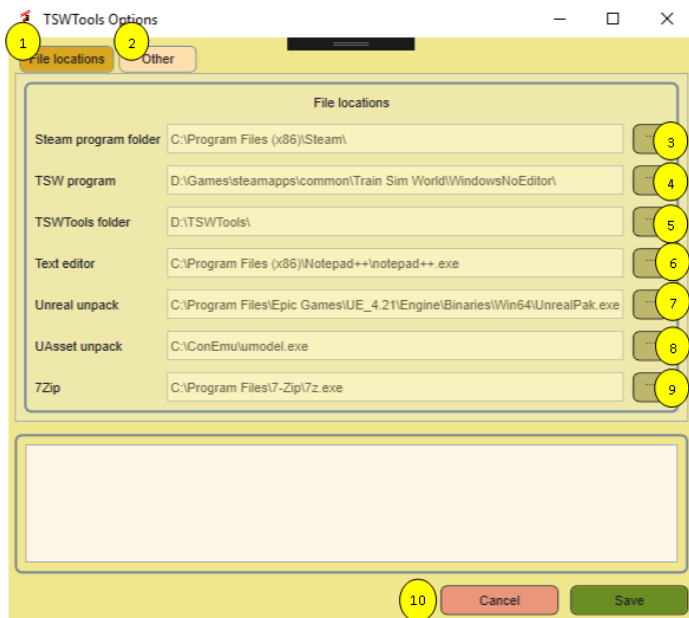
Figure 13 TSW Tools Main Screen



## 5 Utilities

### 5.1 Options Dialog

Before you can use TSWTools you need to set the options. Normally you only need to do this once. Options are stored in the registry.



**Figure 14** Options dialog, File Locations tab.

1. **File locations tab.** The options window is a modal window and it has two tabs. The active tab has a brown colour. The not active tab is red. File locations is the tab for setting all file locations.
2. **Other tab.** The second tab is for other settings.
3. **Steam program folder.** Set here the folder where you installed steam. This is needed for the screenshot manager. This NOT always the folder where steam installs games. In the example you see that at my computer Steam is installed at the C drive, but TSW is installed at the D drive.
4. **TSW Installation folder.** Click at the three dots to open an Open File Dialog. Make sure your path ends with "WindowsNoEditor" otherwise it will not have the desired effect and some functions of TSWTools will not work.
5. **TSWTools folder** is the folder where TSWTools will install its datafiles, e.g. the unpacked game. So this folder requires a lot of space. See section 2.2 for details.
6. **Text editor.** Notepad is used as a default, but I recommend to choose a better option, e.g. Notepad++
7. **Unreal unpacker.** Is the unpacker in the Unreal Engine. If you have the engine installed by using defaults, you probably do not need to do anything. Check out mu website for some additional information on installing the Unreal engine.
8. **UAsset unpack.** Here you need the UModel program. Please check out my website for the download location and additional information.
9. **7Zip.** I think this is not used now, but will be in near future. Here you can provide the location where 7Zip is installed. You need to point to **7z.exe** and not to one of the other programs that come with 7Zip.
10. **OK button** saves your changed settings, **Cancel** will cancel all updates in settings

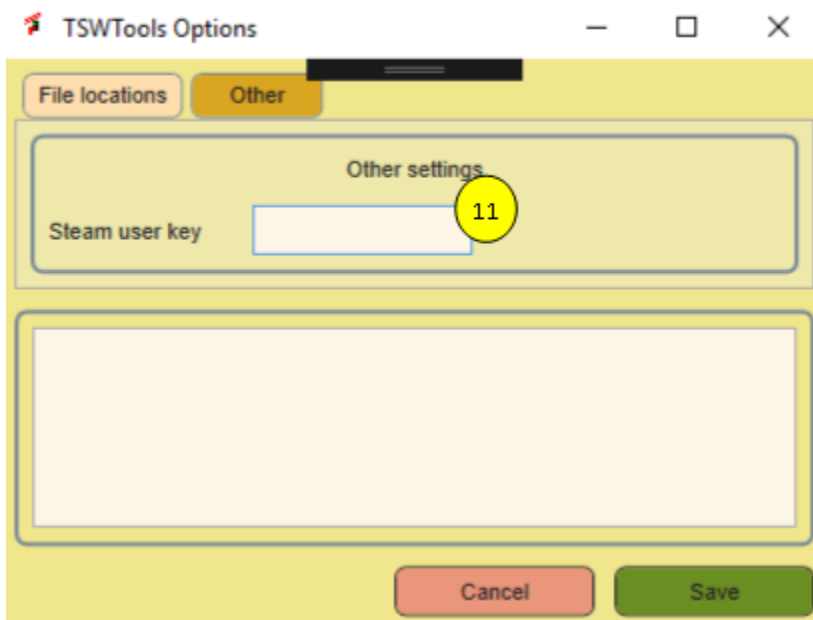


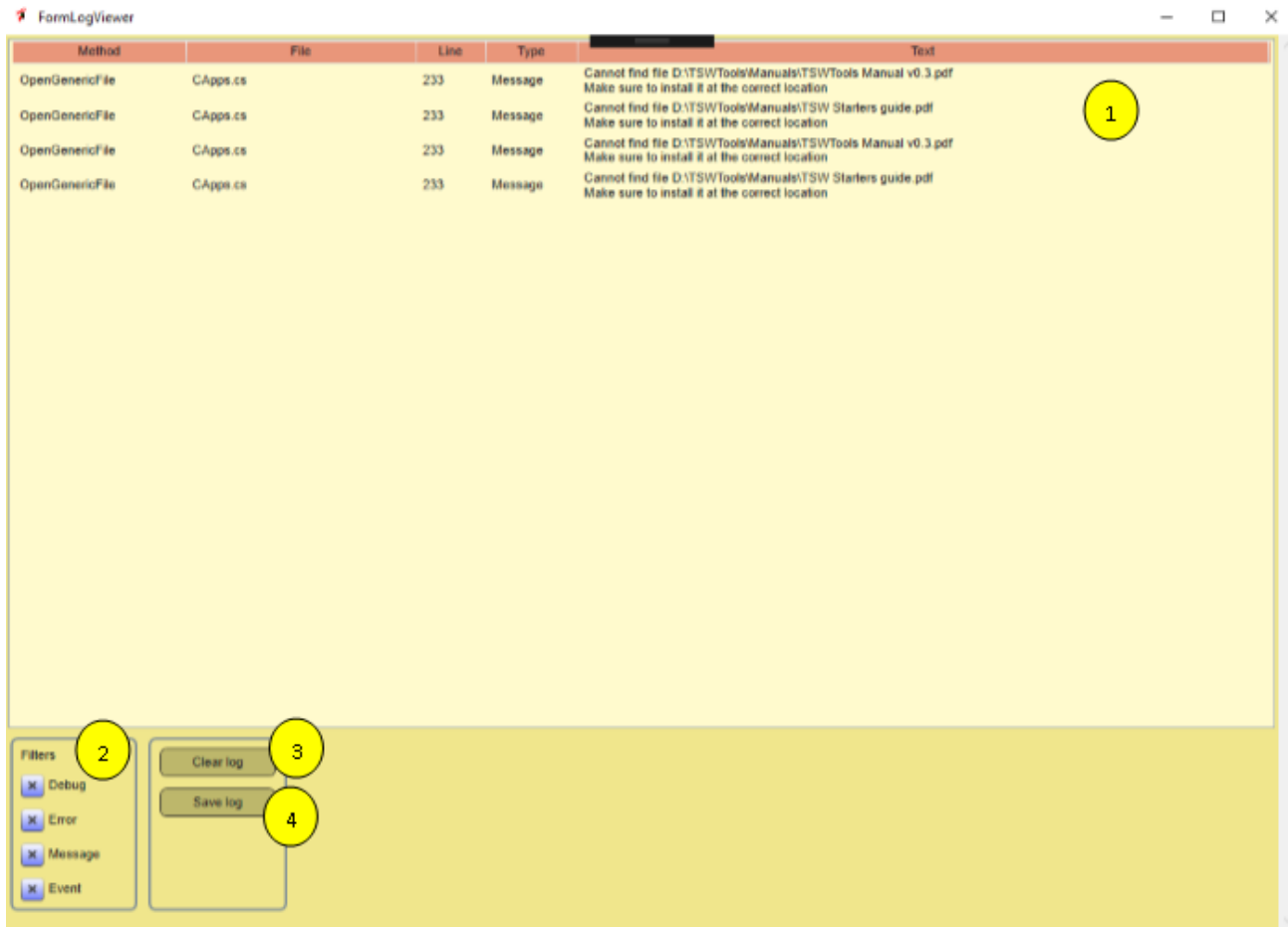
Figure 15 Options, Tab Other visible

11. **Steam User Id.** You need this ID to get the steam controlled screenshots. TSWTools tries to guess the correct id by inspecting your hard disk, but in case it does not work, you can set it manually.



## 5.2 Log viewer

The Log Viewer tells exactly where errors occur and include the error message. It is a non-modal screen and you can leave it open when needed.



**Figure 16 Log viewer**

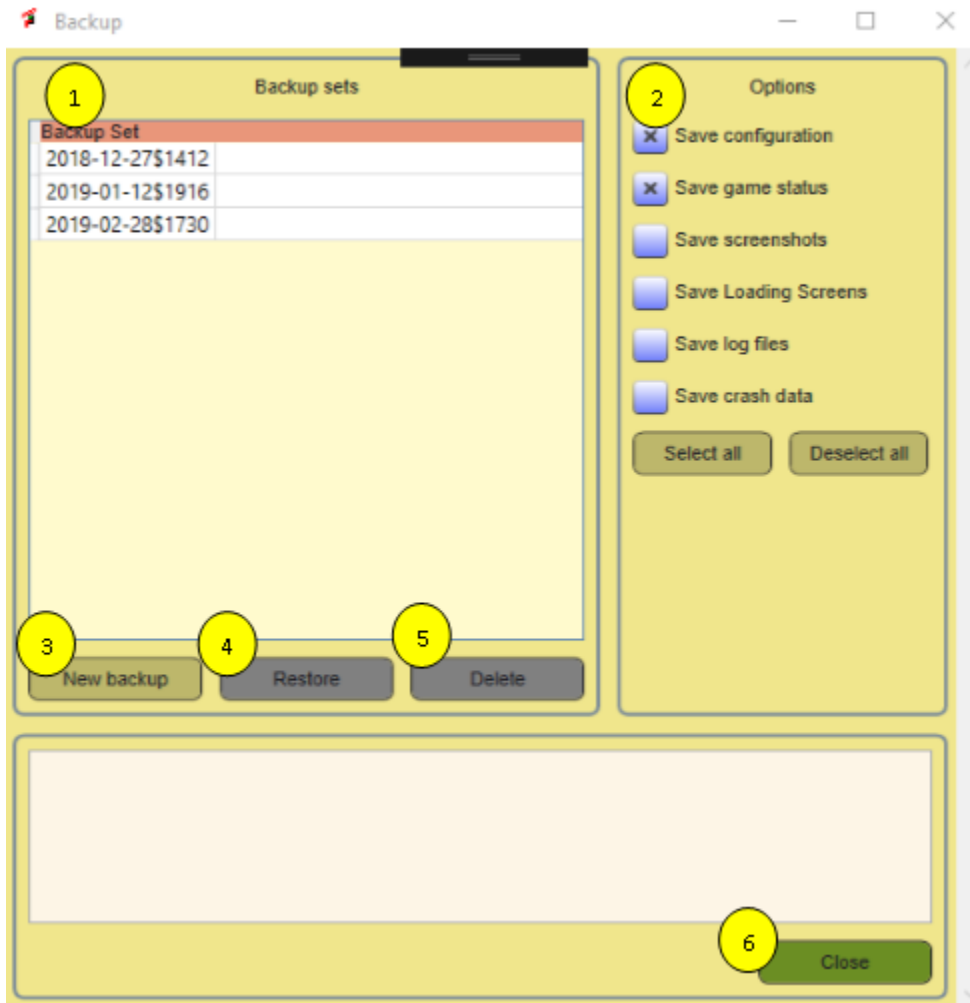
1. Here you find the actual log. It will be updated for each event automatically, no need to refresh it.
2. You can filter what you want to see. Default is all message types on.
3. This button will clear the contents of the log
4. Click here to save the logfile. You will be asked to provide a file name.

**Note:** if you want to report a bug, always send me a logfile. This helps me a lot to find out what is going wrong.

## 5.3 Backup save area

I happened once to me that the saved game data was corrupted. I deleted it and lost all progress. In Figure 17 the backup window is depicted.





**Figure 17 Backup window**

1. This list all previous backups. Each backup is stored in a separate folder, named with the date (yyyy-mm-dd) and the a dollar sign and the system time in four digits. It is likely I will add the option to set a comment text in a next version. So, ask for it if you appreciate that!
2. Options, here you can select what you like to include in the backup. I recommend to use the two checked items as a bare minimum. Optionally you can add screenshots (may consume a lot of disk space!). There are buttons to select the all or deselect them all.
3. Press this button to create the backup. It will NOT ask for further confirmation!
4. If you select an existing backup, you may restore it. Please be warned, you will NOT be asked to confirm this!!!
5. You also can delete backups to free disk space.
6. This closes the window.

## 5.4 View key bindings

Input Mappings

Standard key mapping

Identifier	Action	Key value	Shift	Ctrl	Alt	Cmd	Game pad	Input type
Throttle	IncreaseInputs	A	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gamepad_RightTrigger	StandardInputs
Throttle	DecreaseInputs	D	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gamepad_RightShoulder	StandardInputs
Reverser	IncreaseInputs	W	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gamepad_LeftStick_Up	StandardInputs
Reverser	DecreaseInputs	S	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gamepad_LeftStick_Down	StandardInputs
AutomaticBrake	IncreaseInputs	Apostrophe	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
AutomaticBrake	DecreaseInputs	Semicolon	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
IndependentBrake	IncreaseInputs	RightBracket	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
IndependentBrake	DecreaseInputs	LeftBracket	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
DynamicBrake	IncreaseInputs	Period	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
DynamicBrake	DecreaseInputs	Comma	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
EmergencyBrake	IncreaseInputs	BackSpace	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gamepad_DPad_Down	StandardInputs
EmergencyBrake	DecreaseInputs	BackSpace	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Handbrake	IncreaseInputs	Backslash	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Handbrake	DecreaseInputs	Backslash	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
EngineStartStop	IncreaseInputs	Z	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
EngineStartStop	DecreaseInputs	Z	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Headlights	IncreaseInputs	H	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Headlights	DecreaseInputs	H	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
HeadlightsBack	IncreaseInputs	H	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
HeadlightsBack	DecreaseInputs	H	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Wipers	IncreaseInputs	V	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
Wipers	DecreaseInputs	V	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs
WipersAlt	IncreaseInputs	V	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		StandardInputs

Close

**Figure 18** Key bindings or input mappings

This is not a tool doing much, but it gives you a nice overview of most input mappings. Anyway, it helped me to discover some undocumented features. The mappings are sorted alphabetically. In order to use this function, you need to unpack the game files first.

You can change the sort order by clicking at the column headers. It does NOT reflect the custom changes to keyboard mapping. I do not yet have any clue where this information is stored, probably in the status data, which I cannot open.

**Note:** unfortunately not all key bindings are shown here. Also you can change some of the key binding, and this is not reflected here.



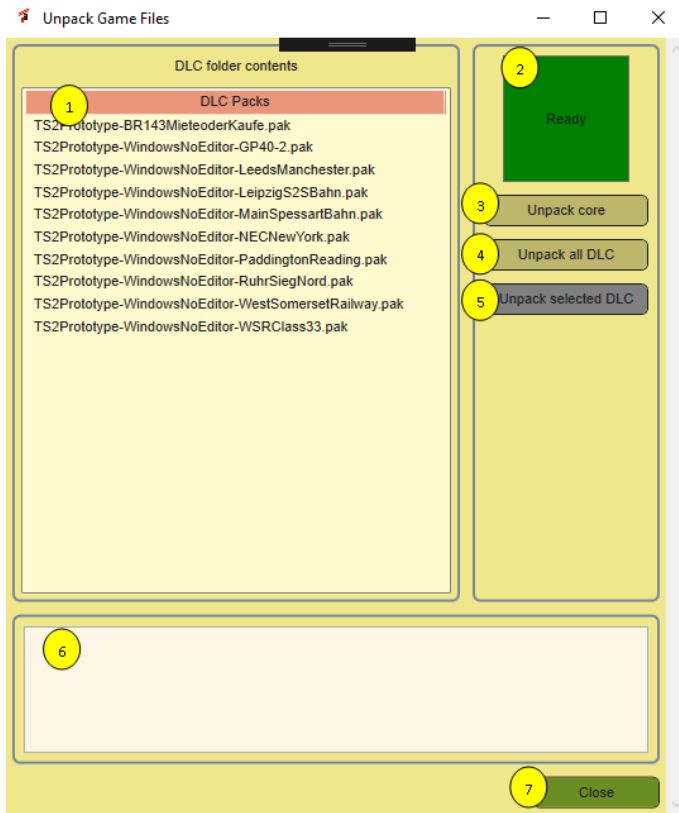
## 6 Unpack and view

### 6.1 Unpack game files

This function will unpack the game files. It cycles through all .pak files it can find and uses the unreal unpacker to do the job. It may take a lot of time, during which TSWTools appear to hang. Please have a lot of patience.

For this function to work, you need to install the Unreal Engine.

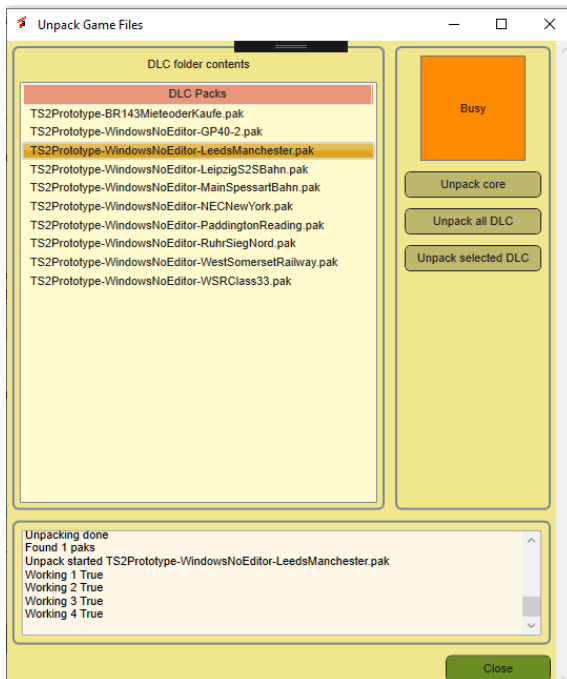
1. Here you have a list of all installed active .pak files. As you may notice, the first one is a livery I have installed now. The game core .pak file is NOT shown here.
2. Because unpacking is time consuming, it is done in a separate process. Every second this process reports is it is still running back to the window. If this square is green and shows the word “ready” it is waiting for an unpack order. During unpacking it will have an orange background and the text “busy”. I would like a more fancy animation, but that is not yet working properly.
3. Click this button to unpack the game core.
4. Click this button to unpack all DLC, **including the game core**.
5. In 1 you can select one or more .pak files. This will enable this button, which does what it says.



**Figure 19** Unpack tool window

6. For the moment you can see here a number that is updated every second, so you have some idea that it is still running.
7. Closes the window.

The unpacked files will be stored in the TSWTools folder. See section 2.2 for details.



**Figure 20** Unpacker while working

## 6.2 View unpacked files

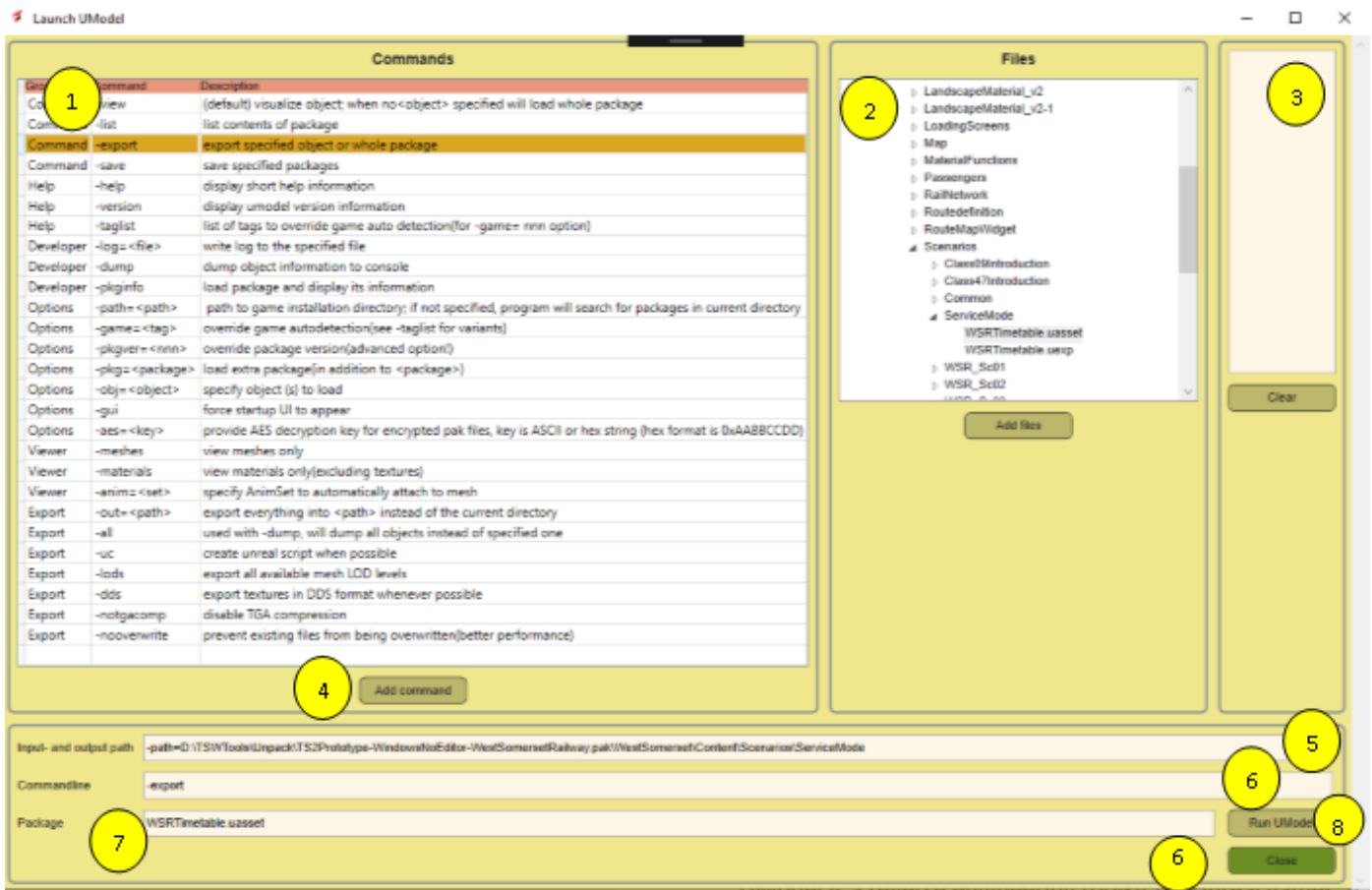
This opens windows explorer, so you can browse through the unpacked files.

## 6.3 Launch UModel

UModel is a reverse engineering toolkit for Unreal games. It has it's own user interface, but I think my solution makes it a bit easier to work with the tool.

**Note:** this function is experimental. Please give me feedback. Until now I have not been able to do anything useful with it.

1. Here you find a list with UModel options that seem most relevant to me, for easier reference.
2. This is a view on thee unpacked assets. You must select a .uasset type file here (likely) and press the Add files button.
3. Here you will find the output of UModel. This textbox will be much wider during execution. I still need to find a better style for this user interface. Coming in the next version...
4. Once you selected an option, this button will insert it at the command line.
5. This line contains the input path you selected in the Files block. You can edit it manually if you like.



**Figure 21** UModel interface for viewing the contents of uasset files

- Here all options you added will be placed. You can (and sometimes must) edit this manually.
- TSWTools will split the path you entered in the Files block and place the filename here. UModel interprets this as the “package”.

The UModel application is called as follows: Input and output path + commandline+ package.

The -out parameter is always set to the path in the TSWTools folder:

```
<TSWToolsFolder>\Unpack\UnpackedAssets
```

Give it a try and let me know please if you have success or if you encounter errors. For your convenience, the UModel help file is included in appendix B of this manual.



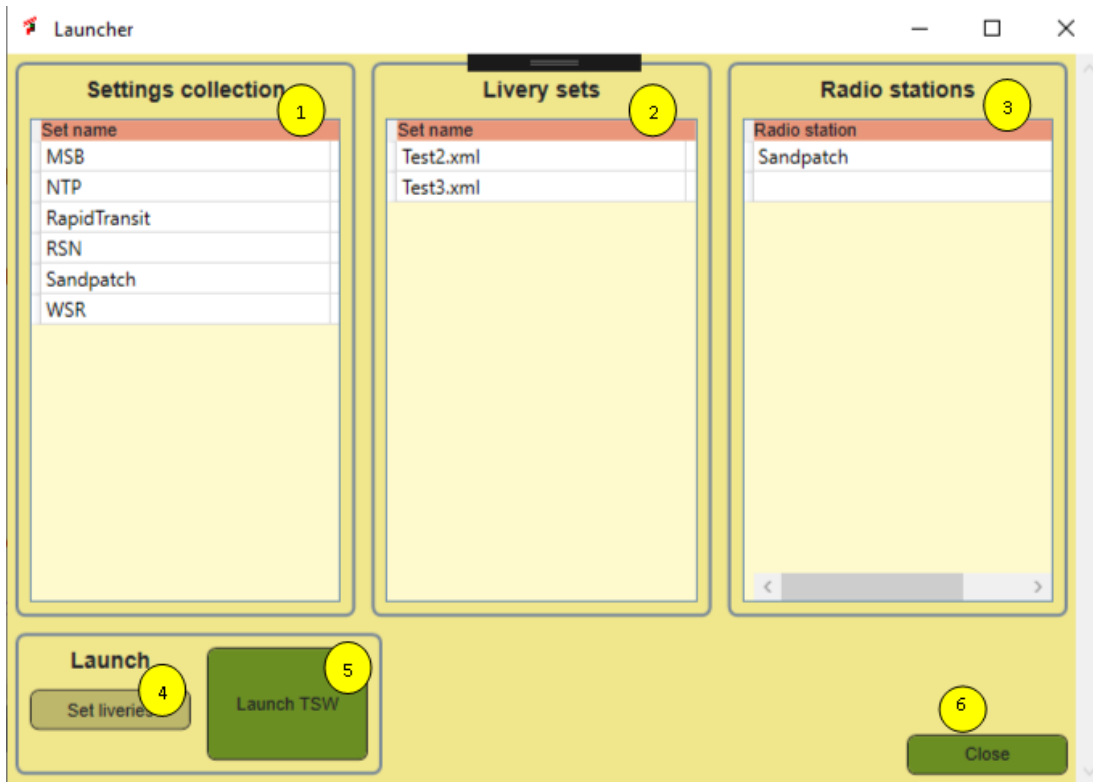
## 7 Tools

### 7.1 Game launcher

TSWTools now has a game launcher. It is useful for two reasons:

1. You can load an options set before launching the game
2. You can activate a railway radio channel
3. You can select liveries, using the liveries manager
4. You can select a predefined set of liveries to use.





**Figure 22** TSW Launcher

In the upper part of the window you see three tables. In each table you can optionally select one row. If you have a row selected, the settings that are represented here will be applied. If you do NOT select a row, nothing will be done with the settings of this type.

**Note:** in the current version, you cannot deselect the items. I will fix this in future.

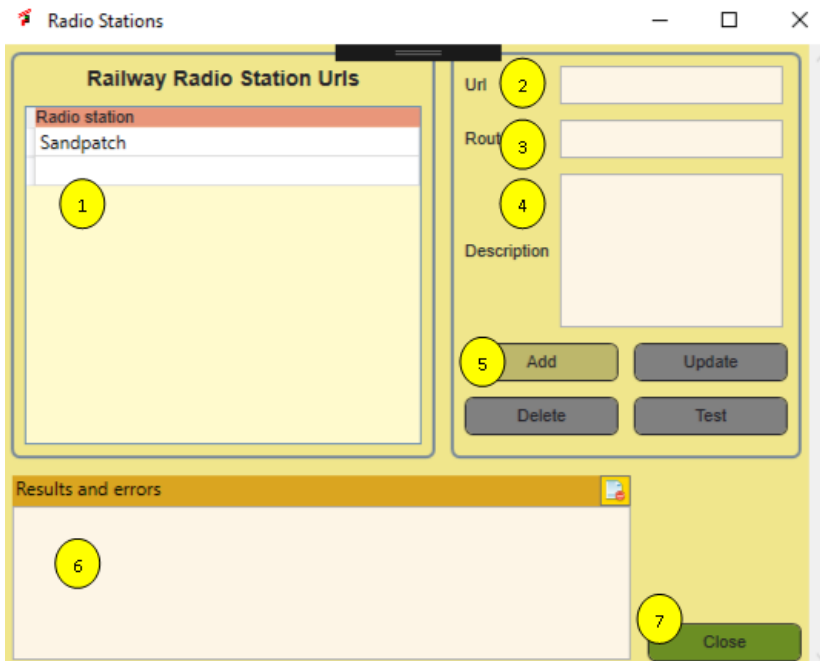
Functions in more detail:

1. Select one of the previously prepared settings files. These are stored in the folder **{TSWTools}\OptionsSets**
2. Here you can apply a complete **livery set**. This is a complex operation, which will be explained in detail shortly.
3. Here you can select a predefined Url for a railway radio station, which will be launched.
4. **Set liveries**, this will open the Livery manager.
5. The **Launch TSW** button. It will set the selected options set, install liveries and activate a radio station. Once this is all done, TSW will be started.
6. **Closes** this window.

It would be nice if we could set some more options, e.g. choose a route or avatar during startup... If you discover any of such functionality, let me know.

## 7.2 Railway Radio Stations

In the previous releases you could open a link to the Railway Radio communications for Sandpatch and run this at the background to increase atmosphere. In this version you can add other radio stations as well. You must use URLs, other ways are not (yet) supported. The list is maintained in the TSWTools database. The window looks like this:



**Figure 23 Railway radio stations window**

Its use is straight forward:

1. A list with all defined radio stations
2. The Url of the station
3. The route name (this will be shown in the list)
4. You may add a description
5. Buttons:
  - a. Add a new Radio station (please make sure to fill the Route field!)
  - b. Update: Update the details of an existing radio station, requires you to select a station from the list
  - c. Delete, delete a station
  - d. Test opens the Url in your default browser
6. Error messages and other issues
7. Closes this window.

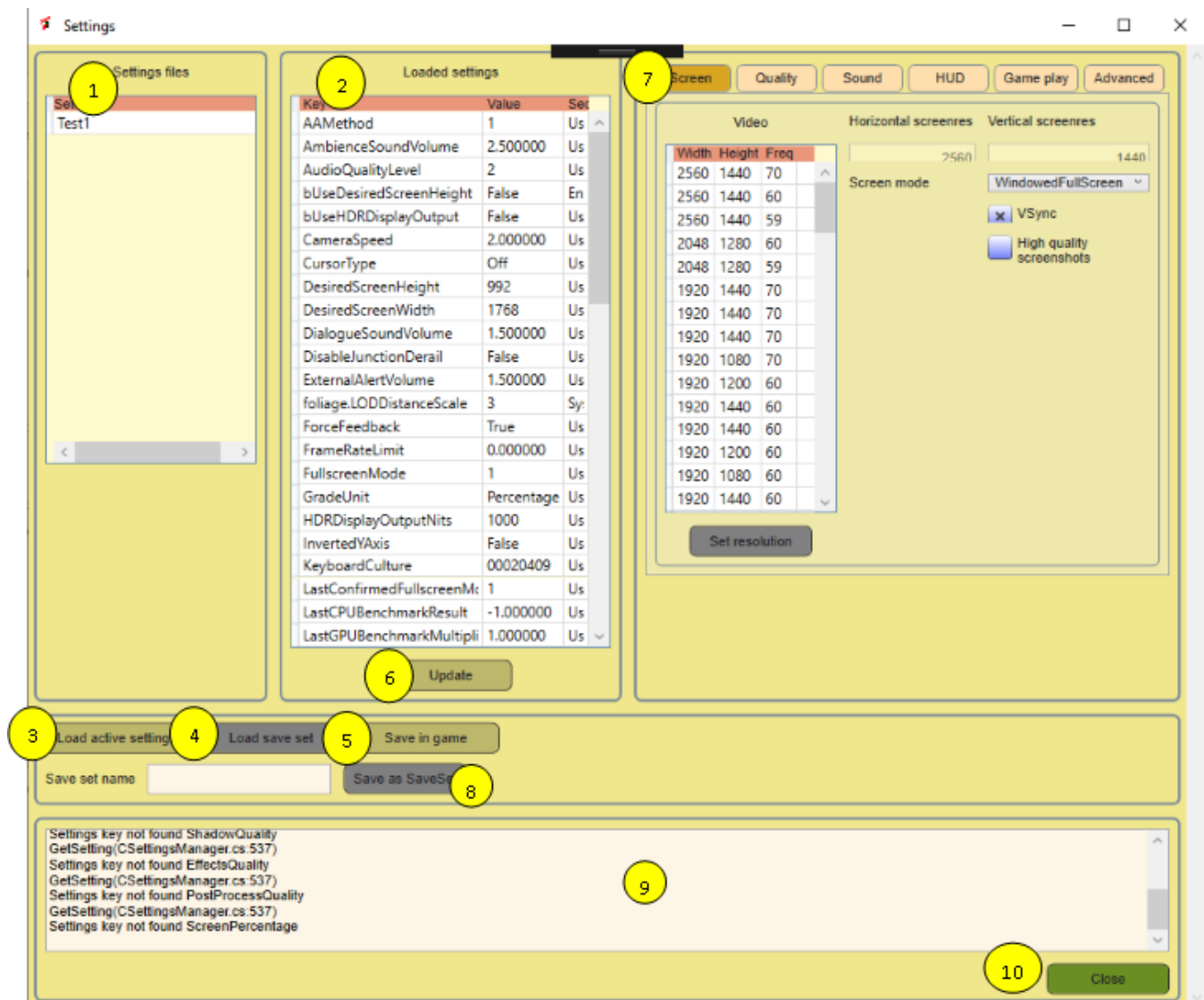
**Note:** I would like to add radio stations to the route guides and I would appreciate to receive links to stations you know about.

## 7.3 Edit TSW Settings

### 7.3.1 Introduction

TSW has a lot of different settings you can adjust in game. This has a number of restrictions:

1. The settings are not always well organized
2. Especially for sound the range for adjustment is too limited. People complain about low sound volumes
3. Some interesting settings are not directly accessible, though you can edit the engine.ini file.
4. It may be useful to have more than one settings set, e.g. one using imperial units and one for metric, a low resolution and a more high resolution set depending on route and how powerful your computer is.



**Figure 24** Settings editor

TSWTools solves this problem for you. It allows to edit most settings, including a number of Unreal settings not covered in game. It allows to save the settings files in the TSWTools folder and load sets in the game during launch. The window is a bit complicated, so please read the instructions carefully. It has a main window and a number of tabs for different types of settings.

The common functions are:

1. A list of all available settings collections (technically each collection is stored in a separate folder).
2. The actually loaded collection of settings in the editor.
3. With this button you can load the settings that are now actually set in game
4. With this button you can load a saved set, you must first select such a set in 1.
5. Saves the edited setting as active game settings
6. Updates the loaded settings from what you changed in the tabs (nr 7)
7. Here you see a number of tabs, each of them will reveal a subset of the editable collection. This will be covered in more detail later.
8. Saves the set you are editing now as a saved set. You must provide a **save set name** in the textbox to enable the button. I require you provide at least three characters in the name.
9. Provides informative messages
10. Closes the settings editor. Note that it will NOT warn you for unsaved changes.

In game, the settings are stored in two different files:

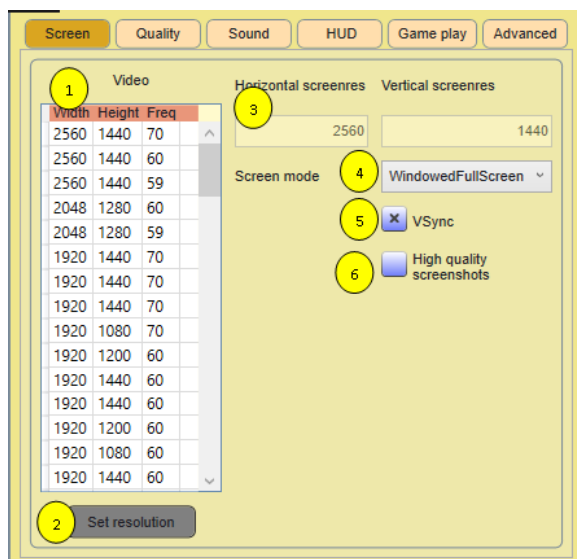
The TSW specific settings reside in **GameUserSettings.ini**. The Unreal settings are stored in **Engine.ini**.

TSW has a bit of strange behaviour. Each setting has a default value. If the default value applies, the setting is not defined in the .ini file. You do not need to worry about this. The Settings function in TSWTools can handle this and knows the default value, but it will always create an entry for the setting. Also settings are organised in groups. TSWTools knows about these groups and will make sure this is working.

If DTG decides to add a new setting, this should not be a problem. You cannot edit it, but its value will be preserved, due to the way this functionality is managed.

In the next sections, all supported settings are explained where necessary. For detailed information on what each setting will do, please read the **TSW Starters Guide**, which is included in the TSWTools installation file.

### 7.3.2 Screen settings



**Figure 25** Screen settings

1. Here you get a list of all screen resolutions your screen supports. The frequency is available here, but it is not used as a game setting. By clicking at the column header, you can have some influence on the sort order.
2. Once selected a resolution, click here to activate it.
3. These two boxes show the set screen resolution.
4. Here you can select the screen mode. It is a combo, so no mistakes possible.
5. Turns vSync on or off
6. Turns high quality screenshots on or off. Warning: high resolution is very high resolution and consumes a lot of disk space.

### 7.3.3 Quality settings

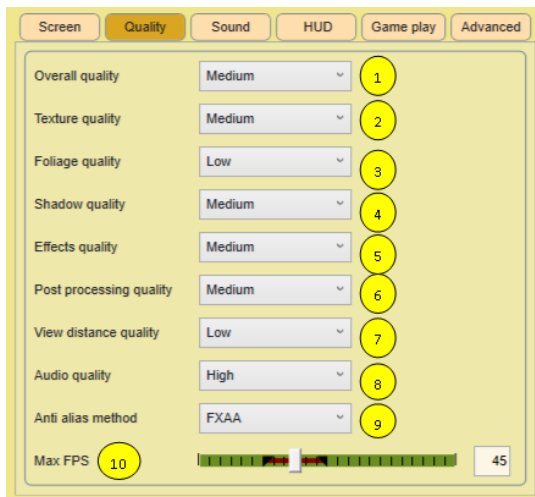
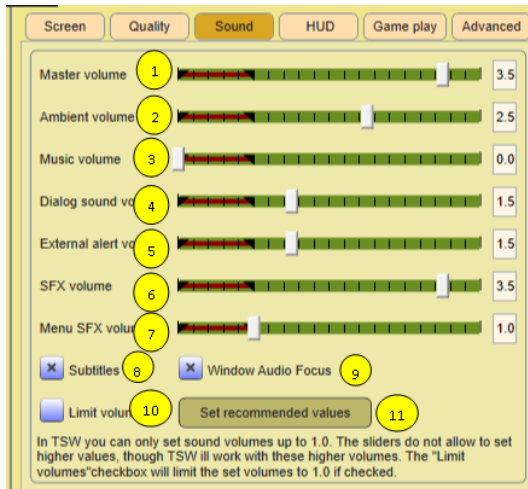


Figure 26 Quality settings

These settings affect the graphics quality. In the **GameUserSettings.ini** file this is not always done consistently. You can select the values Ultra, High, medium and Low for each of them. For the anti-alias method, the technical terms are used. Matt (our DTG expert) recommends FXAA for most cases.

You can limit the number of frames per second. Below 30fps is not a good idea, and 60fps should be enough for high end systems. If you set it to 0, no limit is set.

### 7.3.4 Sound settings



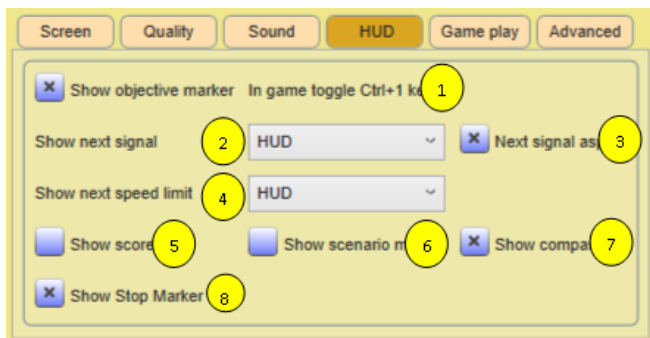
**Figure 27** Sound settings

TSW supports sound settings in the range 0.0 to 1.0. For many users this is not loud enough and higher values are recommended by some users. **You can set these higher values here, but you do it at your own risk.**

In brown you see the officially supported range. If you select a higher value, in the TSW settings menu you can only change it back to the supported range.

- At 8 you turn subtitles on or off.
- At 9 you can set windows audio focus, whatever that means.
- At 10 you can limit the settable sound volumes to the supported volumes.
- At 11 you can set the values to what is considered a good practice.

### 7.3.5 HUD settings



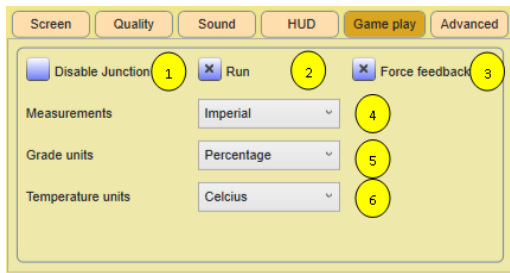
**Figure 28** HUD settings

In TSW you have a lot of control on what you want to see at your screen and how you want to see that. See the TSW Starters Guide for all details. In TSW Tools you can apply the initial settings in a more structured way.

1. The objective marker shows the distance to the next task, either in the form of a marker or included in the HUD. If you set this checkbox, the marker will be shown.
2. The next signal can be shown as marker or in the HUD or both. This setting sets the initial value. You can change it with the keyboard in game. See the TSW Starters Guide for a detailed explanation.

3. You can show or hide the next signal aspect also.
4. Same for the next seed limit, but here there is not an option to hide its value.
5. TSW has a scoring system, you can show the actual score or hide it.
6. TSW has markers in the 3D world where you can start a scenario or tutorial. This setting will hide or show these markers.
7. You can hide or show the compass separately.
8. Sets the visibility of where your train should stop in the 3D world.

### 7.3.6 Game play settings

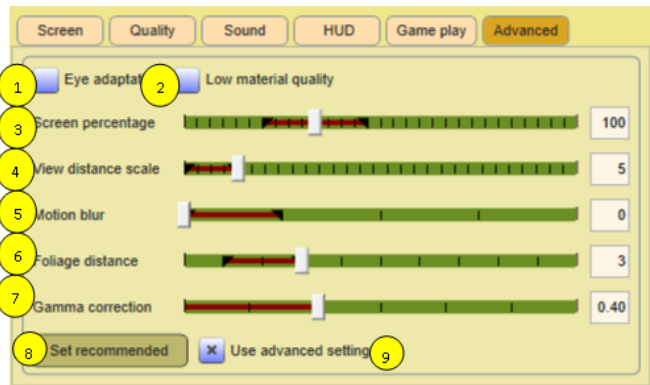


**Figure 29** Game play settings

There a few gameplay settings, which are settable here.

1. Set disable junction derail on or off.
2. Set first person mode default to walk or to run
3. Turn force feedback on or off
4. You can select imperial or metric units (Mph or km/h)
5. You can select grade units
6. You can select temperature units

### 7.3.7 Advanced settings



**Figure 30** Advanced settings

The advanced settings are all settings supported by the Unreal engine. See the **TSW Starters Guide** for detailed information.



For your convenience, at 8 you can select a recommended setting for all parameters. If you deselect 9, the advanced settings will not be used, except for Screen percentage, which will be set to 100%

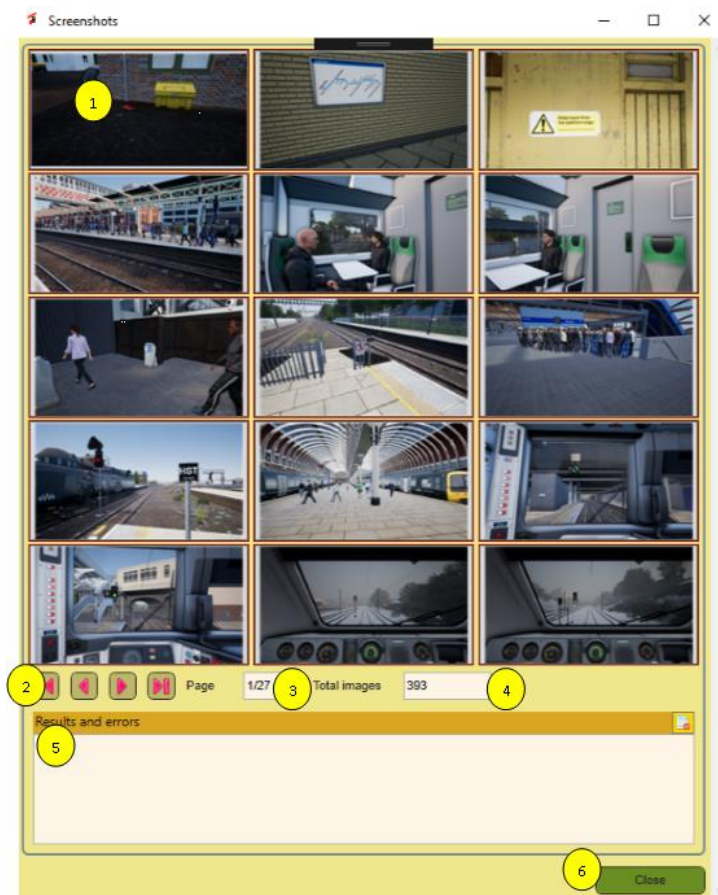
For Motion Blur (5) TSW provides a key combi to turn it on or off. Here you can turn it off or select values in the range 1-4 to determine the amount of motion blur.

Gamma correction (7) makes TSW looks lighter or darker. 0.5 is a good default, you may want to set it slightly higher for night drives.

## 7.4 View screenshots

TSW has essentially two screenshot methods:

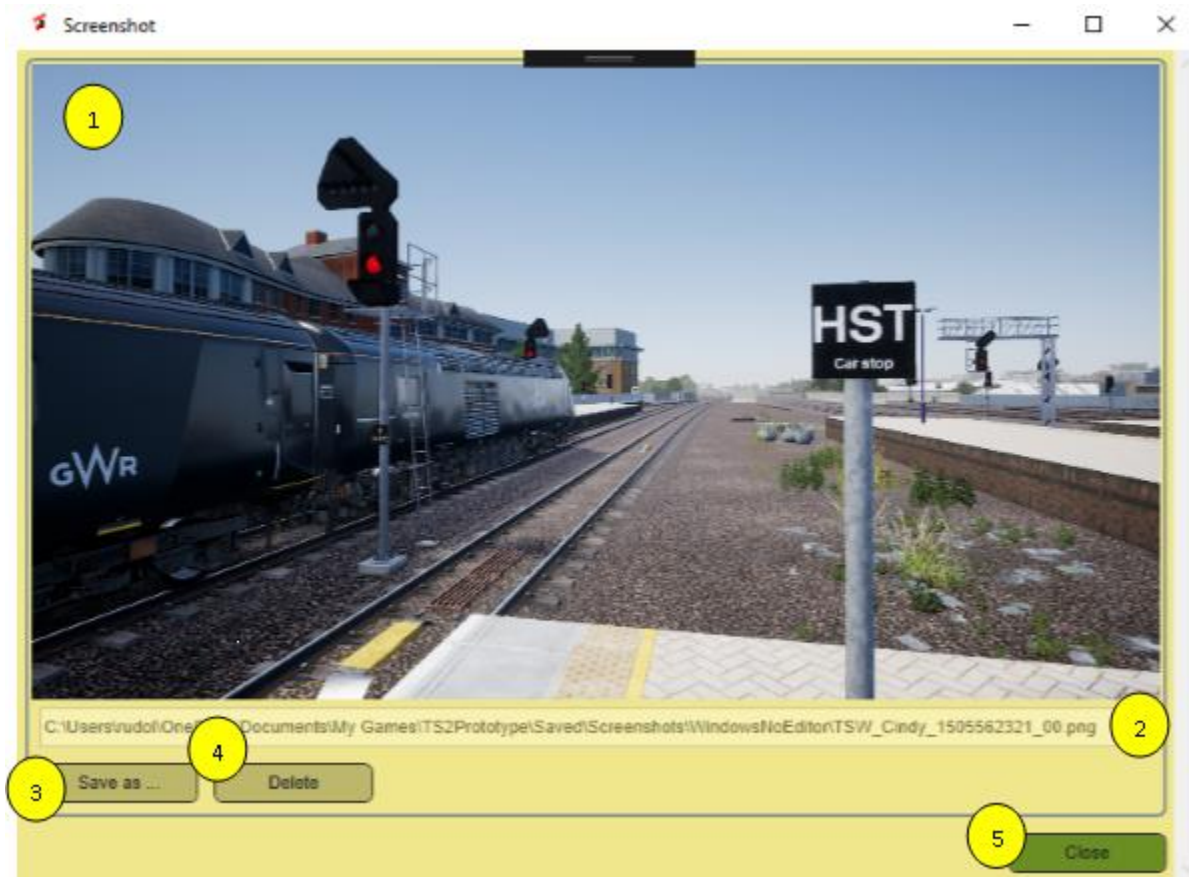
- You can use the F12 key to get a screenshot including the HUD. Essentially this is the steam screenshot facility.
- You can use Ctrl+F12 to get a TSW screenshot, without any HUD stuff.



**Figure 31** Screenshot viewer

Unfortunately the two types of screenshots are stored at different locations and have different size and file format. Therefore TSWTools includes a screenshot viewer that brings them all together. You also can delete screenshots here or make a local copy if you like.

1. Thumbnails of the screenshots, click on a thumbnail to show the screenshot larger!
2. Navigation buttons to go to the first page, next page, previous page or last page.
3. Show at which page you are now.
4. Number of available screenshots
5. Space for error messages
6. Closes this window



**Figure 32 Screenshot preview window**

You can open a separate window for each preview you like to show.

1. Larger preview of screenshot. You can do this by clicking at a thumbnail.
2. File path of the previewed screenshot
3. Opens a file dialog and copies the selected screenshot to the desired location, while updating the file name.
4. Deletes the selected screenshot. Warning: it does NOT ask for confirmation!

1. **Note:** It would be nice to include your screenshots in the loading screen. Unfortunately the loading screens are refreshed when the game starts, so I will not implement such a feature.



## 8 Working with liveries and mods

### 8.1 Introduction

TSW currently has no official features to create your own liveries and mods. However, some people found a way of working that allows minor changes to the game. The good news is that installing them is very easy, just place a file at the proper location. The bad news is that it simply replaces the original object and you can install only one mod as a livery at a time. If you install, say more than one livery for an engine at the same time, it is not so clear what happens. For liveries that cover scenery it is even more unpredictable what happens.

Having said that, TSWTools supports managing liveries as far as possible. You have several options here:

1. Ignore TSWTools and place .pak files in the DLC directory annually and remove them eventually.
2. Use the TSWTools Livery management features, to keep track on the liveries you have and use to the tool to easily install and uninstall liveries and mods.
3. You also can create sets, which combine some liveries and use the TSW Launch Tool to install them specifically at launch time, along with the game settings and eventually Railway Radio Stations.

Along with this comes a **Pak Installer** tool, which supports all three options.

However, it is not a very good idea to mix these options. If you install pak files directly, TSWTools will not know which pak files belong to the game and which are mods, so to be safe, TSWTools chooses the most safe policy, but this may result in unpredictable behaviour.

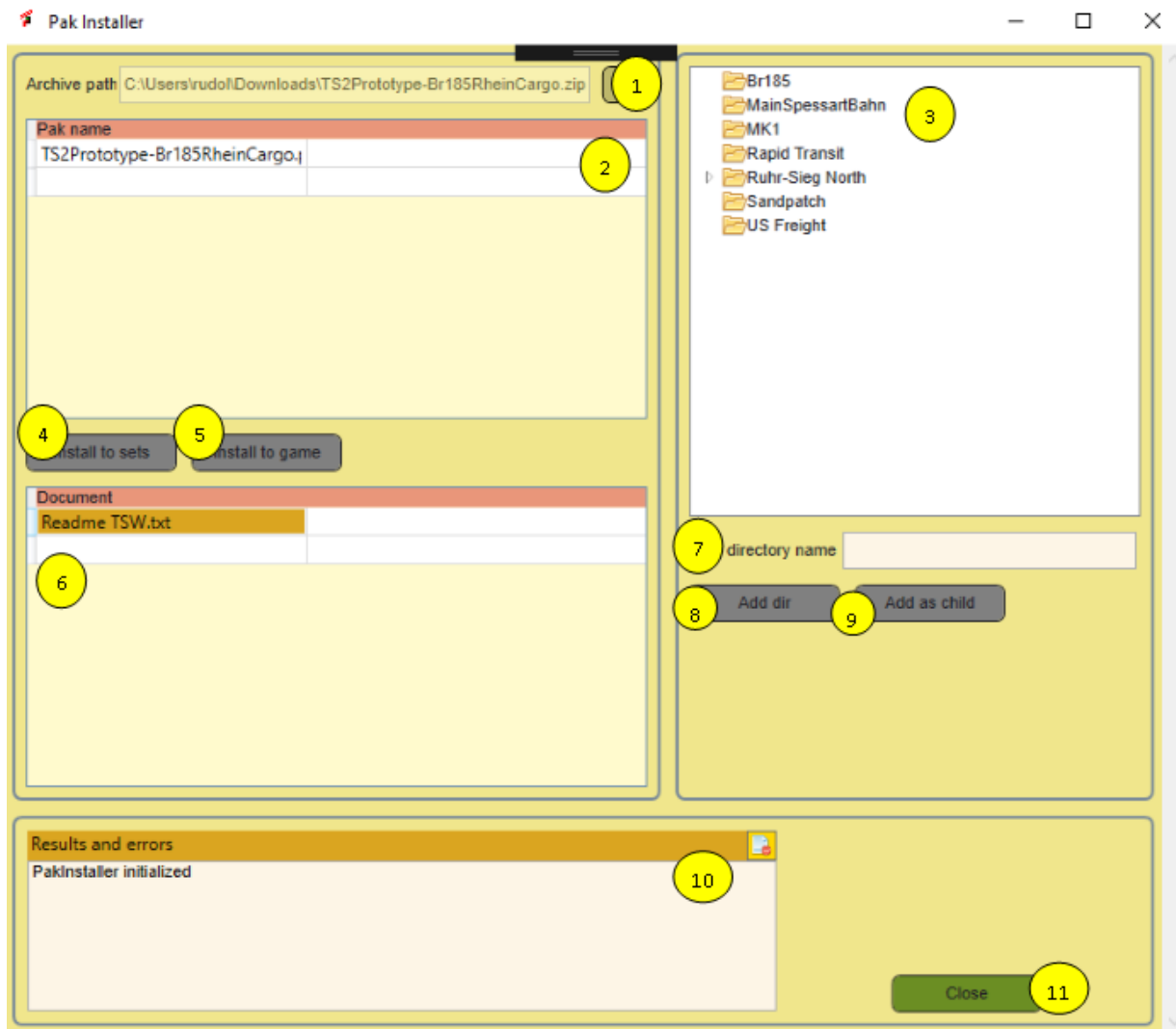
If you want to use TSWTools, do it consistently for the best results. In the next sections the three tools will be described in detail:

1. The Pak Installer
2. The Livery Manager
3. The Livery Sets tool

TSW Launch is explained in section 7.1.

## 8.2 Pak installer

### 8.2.1 Introduction



**Figure 33 Pak installer window**

Installing downloaded add-ons is a bit of a hassle. You need to download them, unzip them somewhere and place the files in the correct location. The Pak installer simplifies this process. You can use the Livery Manager or just skip all this, whatever you like.

The steps are simple:



1. Find the archive.
2. See the .pak files
3. Copy them to the desired location.
4. Optionally, you may review the provided documentation.

Currently, the intention is to use this for livery paks only. In future this may be extended.

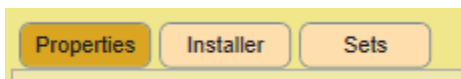
### 8.2.2 Pak installer details

We follow the usual pattern:

1. You need to select an archive first. By default, the Pak installer opens the Download folder and supports **.zip**, **.7z** and **.rar** file format.
2. Here you see the .pak files that are included in the archive. Select the one you want to install
3. You can install the paks to the folder for the livery manager. If you do so, the pak will be registered in the livery database. You **MUST** select a folder here.
4. If you have selected a .pak file (2) and a folder (3) the button will be enabled. It retrieves the .pak file and places it in the specified folder, overwriting existing files without warning.
5. You also can install the pak in the game right away. This option is meant for people who do NOT want to use the livery manager.
6. Here you see all pdf, txt and docx files in the archive. If you click on one of them, it will be opened using the windows shell commands.
7. You may enter a folder name here.
8. If you entered a folder name in 7, you can either add a new folder at the top level using this button.
9. You also can add a folder as a sub folder for an existing folder.
10. Messages from the system go here.
11. Close this window.

## 8.3 Livery manager

TSWTools offers a Livery Manager, that allows you to manage liveries and install or uninstall them with a few clicks. It also maintains a catalogue with all livery packs you have available.



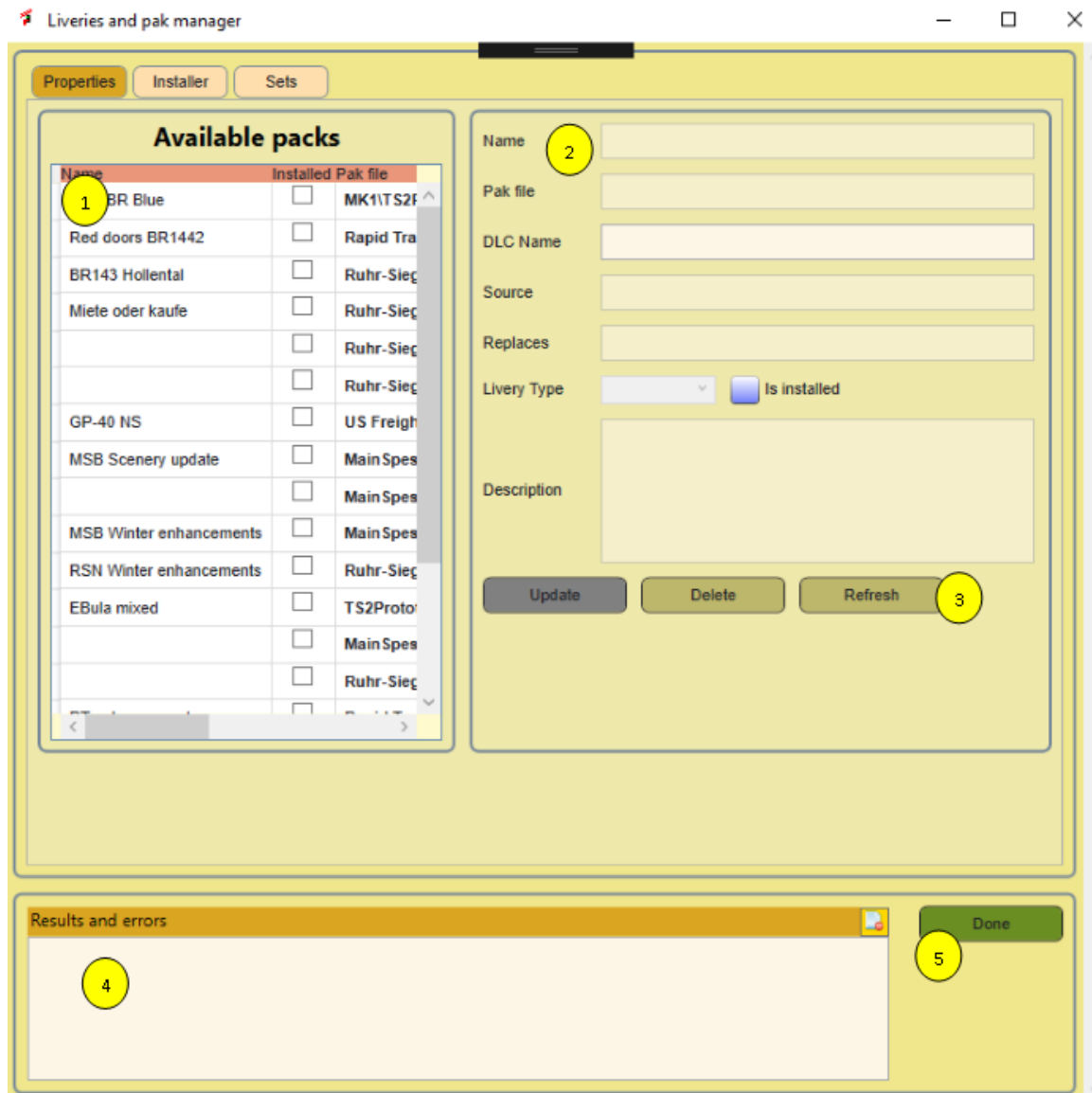
The Livery Manager has three tabs:

1. **Properties** allows you to set metadata for liveries kept in the catalogue. Using this is fully optional, but in near future this will be used, e.g. to enable you to filter liveries.
2. **Installer** allows stored liveries/mod to be installed in the DLC folder of TSW. You also can uninstall liveries easily.
3. **Sets** supports the creation of Livery sets, you like to keep together and install all of them using the installer.

**Note:** At the moment I think it is most convenient to keep these function in a single form. Your opinion and feedback is appreciated.

### 8.3.1 Properties tab

When you start the Liveries manager, TSWTools checks the folder <TSWTools>/Liveries and all sub folders for .pak files. If any are found, the Liveries database is updated. (See section 2.3 for more details). Now you can enhance the data with additional properties. This all is done in the Properties tab of the Livery Manager, which you can invoke from the main window.



**Figure 34** Livery manager Properties tab

A more detailed description:

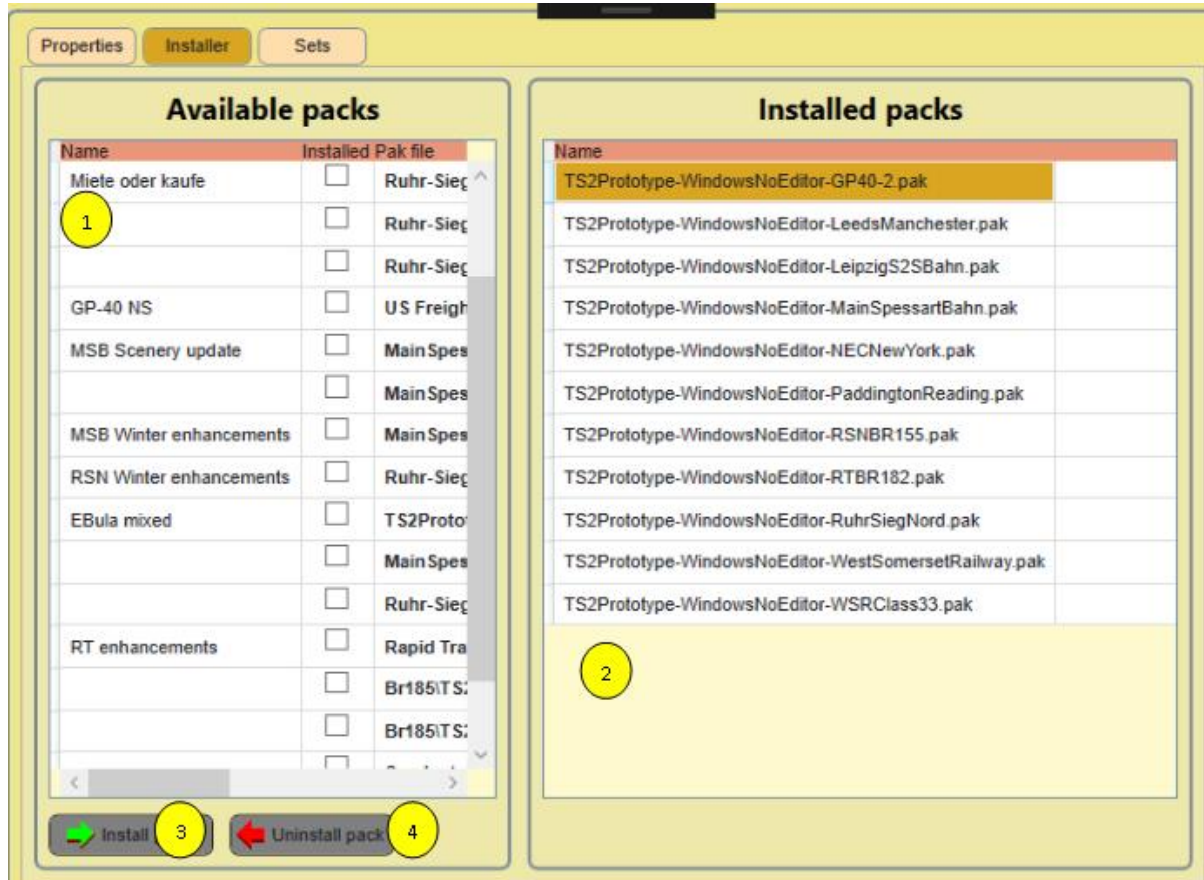
1. A table that reflects the actual liveries registered in the database. Not all columns fit on the screen, but you can scroll.
2. A detailed view showing a single selected record in the database in nr 1. The .pak file textbox and is installed checkbox are read only, you can edit the other fields.
3. The operations possible on the selected record. Update is most important, it will update the database.



4. Shows error messages and other useful things. (Common for all tabs)
5. Closes the window. (Common for all tabs)

### 8.3.2 Install tab

The Install tab covers the functions to actually install or uninstall liveries in the game.



**Figure 35 Livery Manager Install tab**

The controls are explained below:

1. The paks as registered in the database
2. The paks actually installed in the game, this includes the DLC you bought. TSWTools does not know and cannot know which official DLC you have bought.
3. The install button. It is enabled if you selected a pak in nr 1. Press the button and you will see it at the right side in the game. You also will see the checkbox "IsInstalled" is checked.
4. The uninstall button. It requires you to select an installed pak in nr 1. I have built it in this way to prevent you from using TSWTools to delete game files unintended. This button will actually delete the pak file in the game and uncheck the IsInstalled checkbox.

### 8.3.3 Sets tab

The sets tab allows you to create an XML file with a grouped set of liveries/mods. The created sets will be visible in the Launch window and can be used there.

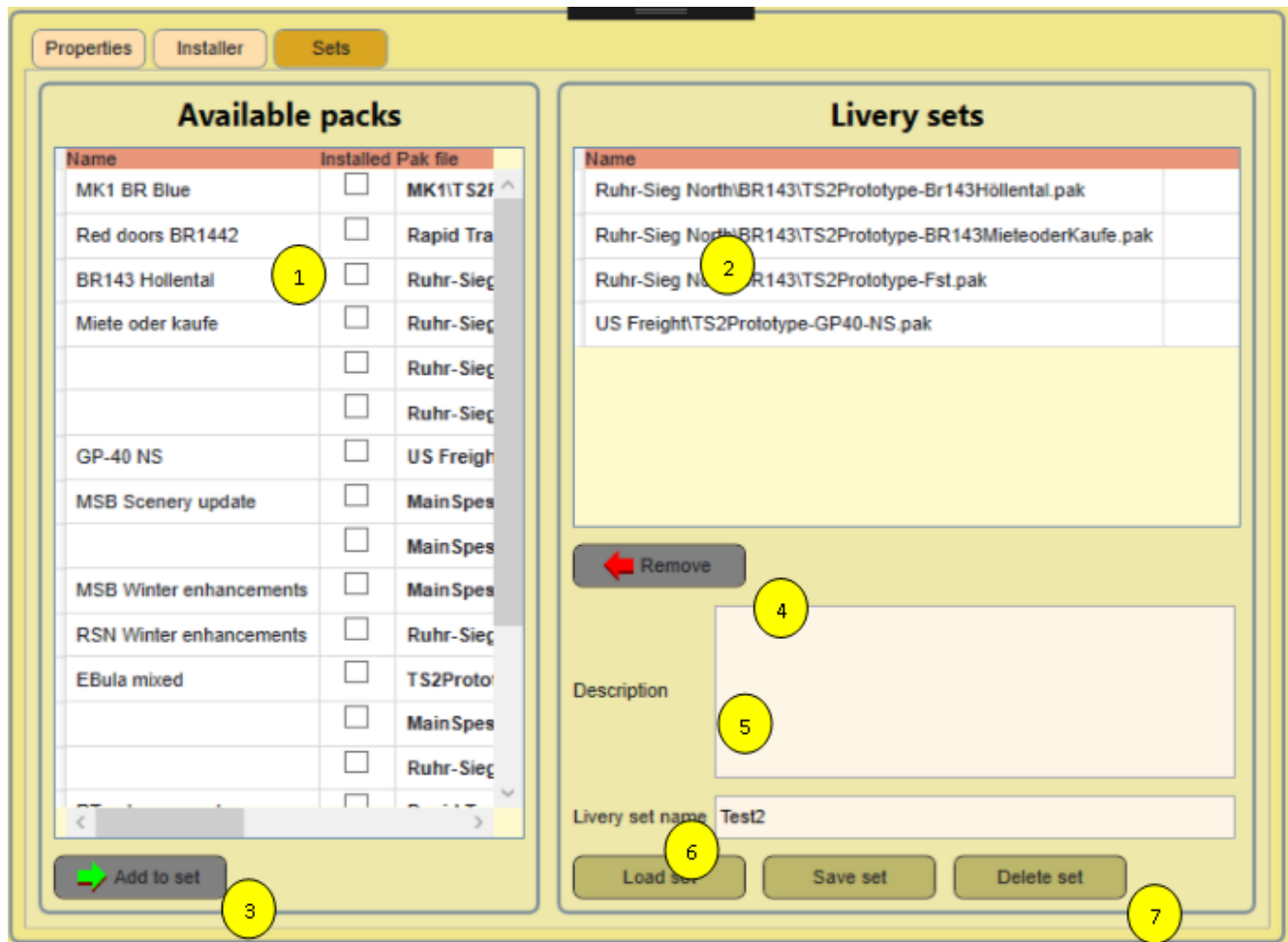


Figure 36 Livery Manager Sets tab

1. Here the same database view as at the other tabs.
2. In this list you will see the paks you selected for the set.
3. Once you selected a pak, you can add it to the set and see it right away in 2.
4. If you select a pak in 2, you can remove it, using this button. The only action is that the pak no longer belongs to the set, nothing else is done.
5. Here you can include a description for the Set. It is not used now in the launcher, but it may be used in future.
6. Here you give your set a name. test2, does not make much sense, but you may use something like **RSN\_Winter**. It is used as a file name, so it must comply these rules.
7. Three buttons that operate on Sets:
  - a. **Load set** retrieves a stored XML file and opens it.
  - b. **Save set** creates the XML file and stores it
  - c. **Delete set** deletes an XML file. It does not impact the data that is visible, so you can use it as a rename function as well.

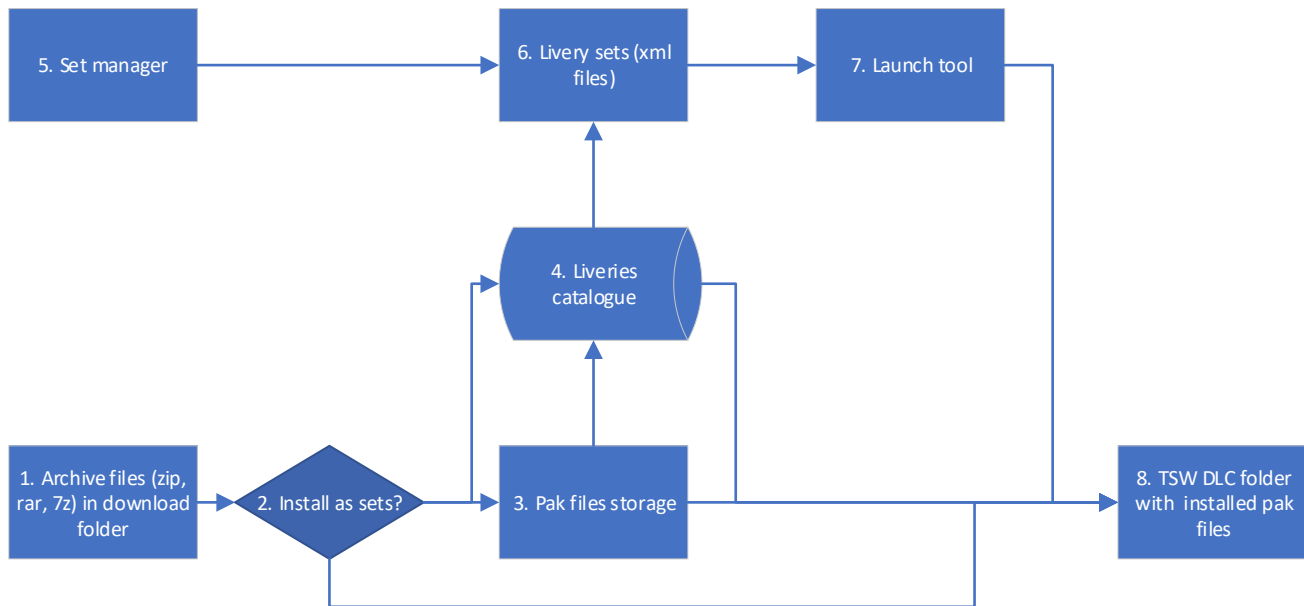
For your convenience, the XML is structured like this:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<PakSet>
  <Description>Some text goed here</Description>
  <PakList>
    <PakName>Ruhr-Sieg North\BR143\TS2Prototype-Br143Höllental.pak</PakName>
    <PakName>Ruhr-Sieg North\BR143\TS2Prototype-BR143MieteoderKaufe.pak</PakName>
    <PakName>Ruhr-Sieg North\BR143\TS2Prototype-Fst.pak</PakName>
    <PakName>US Freight\TS2Prototype-GP40-NS.pak</PakName>
  </PakList>
</PakSet>
```

You may note that a partial path is included, using the Liveries folder as root.

### 8.3.4 Livery manager glued

This all may still be a bit confusing. Let me try to clarify this, using the diagram in Figure 37. Note the boxes are numbered.



**Figure 37 Livery management architecture**

1. It all starts with a downloaded pak file, which is likely to be archived.
2. Using the Pak installer, you can transfer it directly to the game, but then TSWTools will not know what you are doing. So the alternative is to put in the pak files storage (30 and register its existence in the catalogue(4)
3. This is a folder structure, where you can organize all liveries.
4. This is the database, where the liveries catalogue is kept. It also knows, which liveries are installed.
5. The set manager (using the Sets tab) creates XML files containing sets of liveries.
6. These are stored on disk as well.
7. The Launch tool can pick up a livery set and install it. At the same time, it will uninstall all liveries the do not belong to the set, by deleting them from the game.
8. The DLC folder, where the installed liveries reside. Removing them is always done by deleting them!



## 9 Help

### 9.1 About dialog

The About Dialog informs you about the actual version of TSWTools. It also provides a link to the website where you can download updates and additional tools.

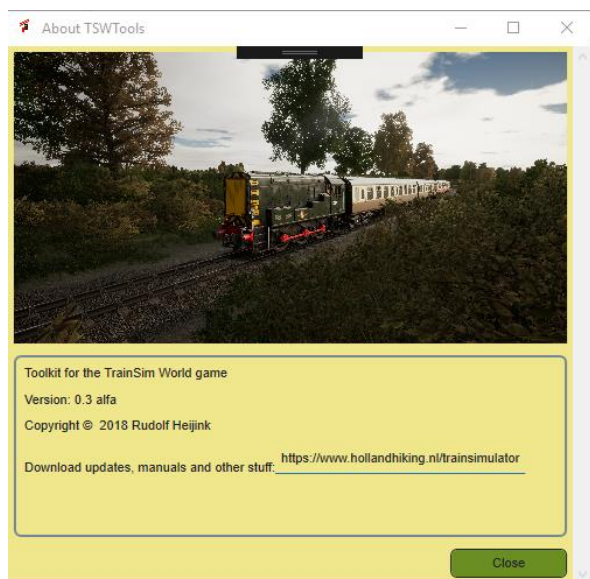


Figure 38 About Dialog

## 9.2 Open TSWTools manual

Clicking the button should open this manual, which is included in the installer. In case you want to install updates, make sure to select the appropriate folder. See section 2.2 for more information.

## 9.3 Open TSW Starters Guide

The TSW Starters guide provides a lot of useful additional game information. A versio will be included in the installer. In case you want to install updates, make sure to select the appropriate folder. See section 2.2 for more information.

## 9.4 Open route guides

In this folder you can install the official game manuals and any additional material as you see fit. Likely, for some routes I will provide additional route guides, which you can download from my website. In case you want to install additional files, make sure to select the appropriate folder. See section 2.2 for more information.

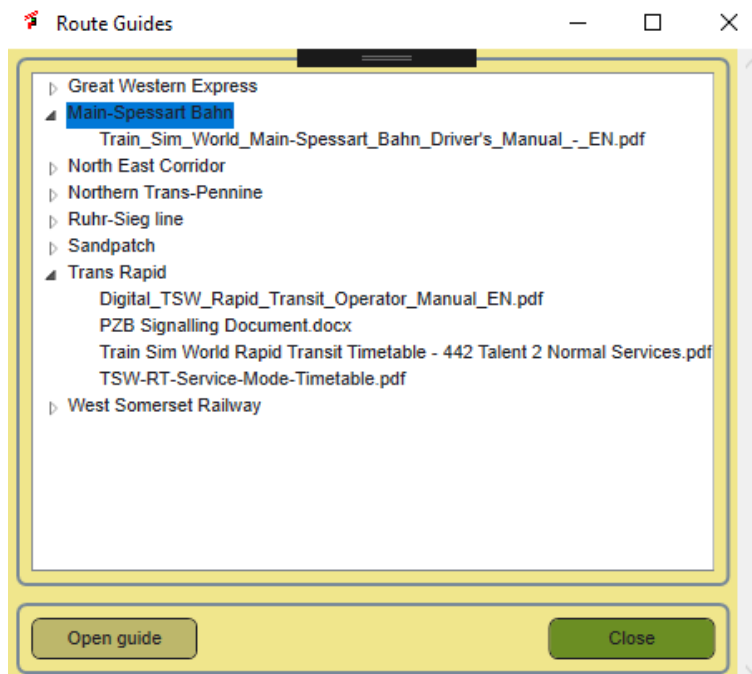


Figure 39 Route guides selector

If you click the button, a simple dialog opens. It represents the folder/file tree for the Manuals/RouteGuides folder. You can select either a directory or a specific document and click the button to open it. TSWTools uses Shell Execute, so it works for all known file types and you are completely free to build your own directory tree.





## A. Download locations

All my guides and tools are available here:

Site name	URL's
Holland Hiking (download page for all tools and guides)	<a href="http://www.hollandhiking.nl/trainsimulator/">http://www.hollandhiking.nl/trainsimulator/</a>

At this site you also will find links to all other tools and manuals mentioned in my guides.

## B. UModel command reference

Unreal Engine viewer and exporter

Link to website:

<http://www.gildor.org/>

```
Usage: umodel [command] [options] <package> [<object> [<class>]]
       umodel [command] [options] <directory>

<package>      name of package to load - this could be a file name
                with or without extension, or wildcard
<object>       name of object to load
<class>        class of object to load (useful, when trying to load
                object with ambiguous name)
<directory>    path to the game (see -path option)

Commands:
  -view         (default) visualize object; when no <object> specified
                will load whole package
  -list         list contents of package
  -export       export specified object or whole package
  -save         save specified packages

Help information:
  -help         display this help page
  -version      display umodel version information
  -taglist      list of tags to override game autodetection (for -game=nnn
option)
  -gamelist     list of supported games

Developer commands:
  -log=file     write log to the specified file
  -dump         dump object information to console
  -pkginfo      load package and display its information

Options:
  -path=PATH    path to game installation directory; if not specified,
                program will search for packages in current directory
  -game=tag     override game autodetection (see -taglist for variants)
  -pkgver=nnn   override package version (advanced option!)
  -pkg=package  load extra package (in addition to <package>)
  -obj=object   specify object(s) to load
  -gui         force startup UI to appear
  -aes=key      provide AES decryption key for encrypted pak files,
                key is ASCII or hex string (hex format is 0xAABBCCDD)

Compatibility options:
  -nomesh       disable loading of SkeletalMesh classes in a case of
                unsupported data format
```



-noanim	disable loading of MeshAnimation classes
-nostat	disable loading of StaticMesh class
-notex	disable loading of Material classes
-nolightmap	disable loading of Lightmap textures
-sounds	allow export of sounds
-3rdparty	allow 3rd party asset export (ScaleForm, FaceFX)
-lzo lzx zlib	force compression method for fully-compressed packages

Platform selection:

-ps3	Playstation 3
-ps4	Playstation 4
-nsw	Nintendo Switch
-ios	iOS (iPhone/iPad)
-android	Android

Viewer options:

-meshes	view meshes only
-materials	view materials only (excluding textures)
-anim=<set>	specify AnimSet to automatically attach to mesh

Export options:

-out=PATH	export everything into PATH instead of the current directory
-all	used with -dump, will dump all objects instead of specified one
-uncook	use original package name as a base export directory (UE3)
-groups	use group names instead of class names for directories (UE1-3)
-uc	create unreal script when possible
-psk	use ActorX format for meshes (default)
-md5	use md5mesh/md5anim format for skeletal mesh
-glTF	use glTF 2.0 format for mesh
-lods	export all available mesh LOD levels
-dds	export textures in DDS format whenever possible
-notgacomp	disable TGA compression
-nooverwrite	prevent existing files from being overwritten (better performance)

Supported resources for export:

SkeletalMesh	exported as ActorX psk file, MD5Mesh or glTF
MeshAnimation	exported as ActorX psa file or MD5Anim
VertMesh	exported as Unreal 3d file
StaticMesh	exported as psk file with no skeleton (pskx) or glTF
Texture	exported in tga or dds format
Sounds	file extension depends on object contents
ScaleForm	gfx
FaceFX	fxa
Sound	exported "as is"

For list of supported games please use -gamelist option.

For details and updates please visit <http://www.gildor.org/en/projects/umodel>

## C. Known issues

There are some issues I am aware of, but not yet solved. These are listed here.

Issue nr	Description	Priority
1	On small screens the user interface is not always looking good	Medium
2	The UModel interface is still experimental	Medium
3	The index for this manual should be updated	Low
4	In combination with OneDrive loading screenshots performs poor	Medium
5	Thumbs are not properly updated when you delete a screenshot	Medium

## Index

About Dialog .....	47	Notepad++ .....	10
download link .....	37	options .....	18, 20
download links.....	10	screenshot .....	37
input mapper .....	18	text editor .....	10, 21
input mapping .....	24	TSW folder .....	20
Installation .....	9	TSWTools folder .....	21
key binding .....	18, 24	unpack.....	25
main screen .....	18	Unreal game engine.....	10
manual.....	10	Unreal unpacker .....	18, 21
New features .....	9		