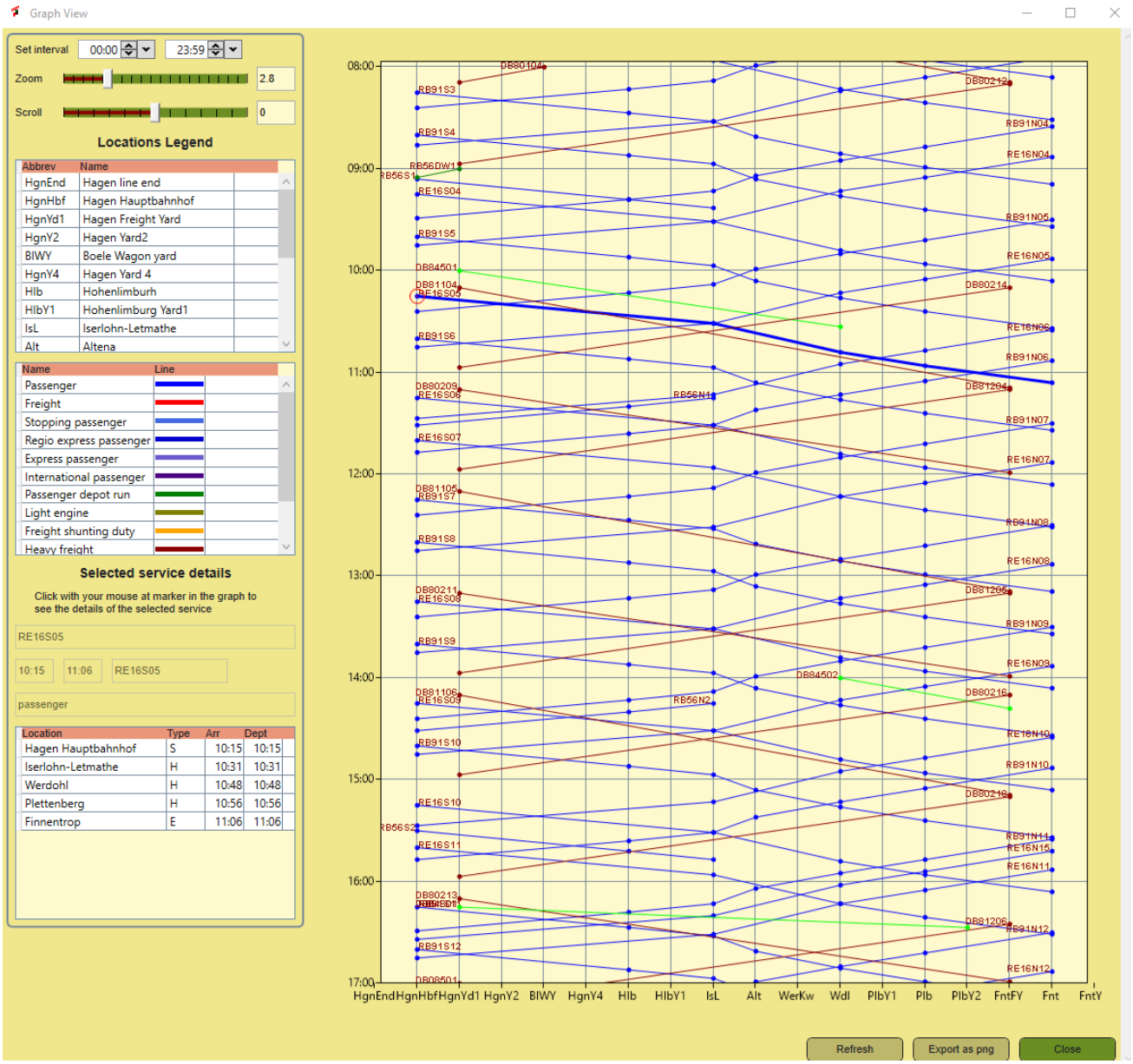


TimetableTool Manual

Rudolf Heijink

Version 0.4.0, August 2020

Copyright © 2020 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.
- Dovetail Games has been granted a non-exclusive license for commercial use of this guide.

Preface

Introduction

Images

Each chapter is introduced with a screenshot I made in game.

Acknowledgements

I am very grateful to Scott Harden, for developing a simple to use graphics library.
https://swharden.com/scottplot/ and for his advice to make the graphs look better.

Special thanks to William Pannone for his early feedback and patience with the bugs that are still there.

A big thank you to all community members that contributed to the contents of this manual and the tool. Many of them will not be aware of their contribution, but lots of information comes from the community forums.

Frontpage image:

ScottPlot Graph view, one of the available timetable reports.

Disclaimer

This manual and TimetableTool are provided “as is” the author cannot accept any consequences from the use of Tool and Manual.

The contents is the sole responsibility of the author.

Contact

Comments are welcome at trainsimulator@hollandhiking.nl.

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

If you volunteer to help me developing TimetableTool please contact me!

Enjoy reading!

Rudolf Heijink

Contents

Preface **4**

1 Introduction **8**

1.1 Warnings 8
1.2 Donations 9

2 Updates **10**

2.1 New in this version 10
2.2 Next version 10
2.3 Versions in development 10
2.4 Version 0.4.0 10
2.5 Version 0.3.0 11
2.6 Version 0.2.0 11
2.7 Version 0.1.0 11

3 Installation **12**

3.1 Basic installation 12
3.2 Deprecated database 12

4 About timetables **13**

5 Creating a Timetable **16**

5.1 Define a route 17
5.2 Locations 18
5.3 Service directions 20
5.4 Service templates 21
5.5 Services 28
5.6 Timetables 29
5.7 Show the timetables 31

6 Reference guide **32**

6.1 Locations 32
6.2 User interface 33
6.2.1 Menu 33
6.2.2 Information panel 34
6.2.3 Tables 35
6.2.4 Buttons 36
6.2.5 Editor 37
6.2.6 Services editor 37
6.3 User settings 39
6.4 About window 39
6.4.1 Manual 40
6.4.2 Logging 40
6.5 Viewing ScottPlot graphs 41
6.6 The database 43
6.7 Backup and restore 45
6.8 Export and import of routes 46
6.9 Source code 47

7 Known issues **48**

A. Links to documentation and tools **49**

List of figures

Figure 1 Classic timetable view as generated by TimetableTool 13
Figure 2 Example time graph for West Somerset Railway. 14
Figure 3 Graph view of the complete services timetable for West Somerset Railway 14
Figure 4 Arrivals and departures for a selected location 15
Figure 5 Route selection and entry screen. 17
Figure 6 Completed Route edit form 18
Figure 7 A route is selected, which enables some other screens, e.g. locations 18
Figure 8 Route layout as shown in the official game manual 19
Figure 9 Completed location list for the RSN route. 20
Figure 10 Service direction table filled for RSN 21
Figure 11 Service templates form 22
Figure 12 Part of the North-South passenger timetable for RSN 22
Figure 13 RB56 service defined and time events loaded 23
Figure 14 Completed time events for service RB56 25
Figure 15 Saved time events 26
Figure 16 RB91 en RE16 services 27
Figure 17 Two of the three northbound services 28
Figure 18 Define services 28
Figure 19 Timetable window 29
Figure 20 Selected timetable showing available services 30
Figure 21 Reports screen 31
Figure 22 Example of a complex network 32
Figure 23 Menu bar 33
Figure 24 Information panel 34
Figure 25 Tables example 35
Figure 26 Table with selected row 36
Figure 27 Buttons below a table 36
Figure 28 Editor part 37
Figure 29 repeat function for services 38
Figure 30 User settings dialog 39
Figure 31 About window 39
Figure 32 Popup window 40
Figure 33 Log viewer 40
Figure 34 Zoomed graph view 42
Figure 35 Timetable graph for Ruhr-Sieg Nord 42
Figure 36 Timetable graph with a selected service 43
Figure 37 Database structure 44
Figure 38 Backup window 45
Figure 39 Export and import the TimetableTool database 47

1 Introduction

TrainSimWorld has a big new feature, Timetable services, where you can choose from up to 350 drives on a single day. In the present version, there is only one timetable, but I believe some day DTG will make it possible to develop our own timetables. West Somerset Railway may be a good example, what about this:

- A day with only some maintenance activities
- A diesel heritage day
- Separate summer/winter time tables

And whatever you can think of. Designing all services for such a day, requires new skills and tools. I believe a tool to plan your services ahead and plot them into one or more timetables is absolutely necessary. You simply cannot test 25 one hour during services properly.

Therefore, I have the idea of creating a timetable tool for some time. Now I take the opportunity to combine this with learning a lot of new C# programming skills.

At present, we arrived at version 0.4 Sill alpha and it definitely has some bugs. But there are more report types and the reports look better. Development will continue to support the Scenario Planner for TSW2 specifically.

I included two completely worked out timetables:

The services timetable for Heavy Haul (Sandpatch) and the timetable for WSR. This shows how it works and how to use it, while being simple enough to reveal errors fast. These two timetables will be installed automatically. Some more timetables for TSW can be downloaded from my website.

1.1 Warnings

You can use TimetableTool for any game timetable. It is NOT intended to create timetables for real world application and is probably not suitable for that purpose. I cannot accept any liability if you try to use this for anything in a real world.

The first version is only meant to get your feedback. It may not be compatible with future versions and you may need to retype all data, though I will try to avoid such a situation.

1.2 Donations

Until now I never asked for donations for my software. I do not need to make money with them, but as I get more experience, and applications get more complex, my expenses are increasing. For example, I use some development tools that are not free and need to do some additional courses etcetera.

Finally, I will need to purchase code signing certificates. This makes sure you can trust my software.

Therefore, now ask you to donate me if you sue this software:

- For professional developers, I ask **Euro 25 per year**. This is cheap
- For non-commercial use, a one time **Euro 5** donation makes me happy.

You can use this link, the QR code, or use the paypal link at my website.

https://www.paypal.com/cgi-bin/webscr?cmd=_donations&business=LNBS2R49HHBF6&currency_code=EUR&source=url

2 Updates

2.1 New in this version

Highlights for this version:

- An improved graph using the ScottPlot library.
- First, very simple user settings page
- Removed the old graph from the code.

Bug fixes:

- Updated the example timetables
- Fixed a bug, that makes it impossible to edit a single time event
- Fixed a bug that caused counting errors when adding services t a timetable

2.2 Next version

There definitely will be some improvements to make it all work a bit more smoothly.

I may add two new functions:

1. A timetable at train level, so you can decide/follow where each train is. This would make the while timetable more realistic.
2. I also would like to add some statistics reports, e.g. the number of services with details for service class, direction and maybe some other views.

I think that is quite lot for now. I also may have a surprise feature for you …

2.3 Versions in development

The source code is publicly available for free at Github. There you can see the most recent versions, but keep in mind they may have bugs and you need to compile the code by yourself. If you want to help me, let me know!

Technologies used:

1. SQLite database
2. Visual Studio 2019 Community Edition, Net Core3.1
3. C# 8.0
4. ScottPlot graphics library
5. Caliburn.Micro MVVM library
6. Dapper for Database Access
7. Inno Setup
8. Word and PDF for the user manual

2.4 Version 0.4.0

Highlights for this version:

- An improved graph using the ScottPlot library.
- First, very simple user settings page
- Removed the old graph from the code.

Bug fixes:

- Updated the example timetables
- Fixed a bug, that makes it impossible to edit a single time event

- Fixed a bug that caused counting errors when adding services to a timetable

2.5 Version 0.3.0

- A new report which shows for a specific location departure and arrival timetables.
- I rearranged the different views on a timetable into one form. This makes it a bit easier to use, you no longer need to open two different screens. This feels more natural.
- Service types now must be selected from a list. This is a preparation for some improvements to the graphs.
- You can now delete data from the database

Bug fixes:

- Fixed an installer issue that made it sometimes impossible to run TimetableTool.
- Localization issue: in the service editor you now MUST use the 24 hour time format. Using a localized format might cause a crash.
- Resolved an issue with the delete function for routes

2.6 Version 0.2.0

This version has some new functions:

1. Backup and restore of the database.
2. Export and import of data at route level.
3. Delete functions are now working (may require some manual action).
4. Renaming, services renamed to service templates and service instances to services. This was kind of a huge task, but I feel this makes more clear how this all works.
5. Two new shortcuts:
    a. You now can copy data from the service template to create a service
    b. You can repeat a service, if it is run more than once. This is a complex function, so read the manual before using it.

2.7 Version 0.1.0

This version is the initial version. It is not meant for operational use. Just for review.

3 Installation

3.1 Basic installation

TimetableTool comes with a installer. Just run it. It will install a 32-bit application in your programs directory.

It also will create a new folder, named TimetableTool in your Documents folder. This folder contains this manual and the TimetableTool database.

The database contains two TrainSimWorld timetables:

1. The West Somerset Railway
2. Heavy Haul (Sandpatch)

This allows you to get started right away.

You will not yet get updates automatically. That is on my wish list, but will take some time.

Both demo routes are updated. These updates will not be installed automatically. You will find them in your Timetable data folder. You can delete the installed versions and the import the demo routes manually.

There also aresome more routes with timetables for TSW you can download from the website and import.

3.2 Deprecated database

It is possible that you get this error message:

```
Deprecated database version. Check user manual ch3.2 for a solution.
```

This mean that I created a new database, which is not compatible with the version you use. Solving the issue is not very hard.

1. Export all the routes you need to preserve.
2. Delete the active database (n worry, a backup will be created)
3. Import the routes again.

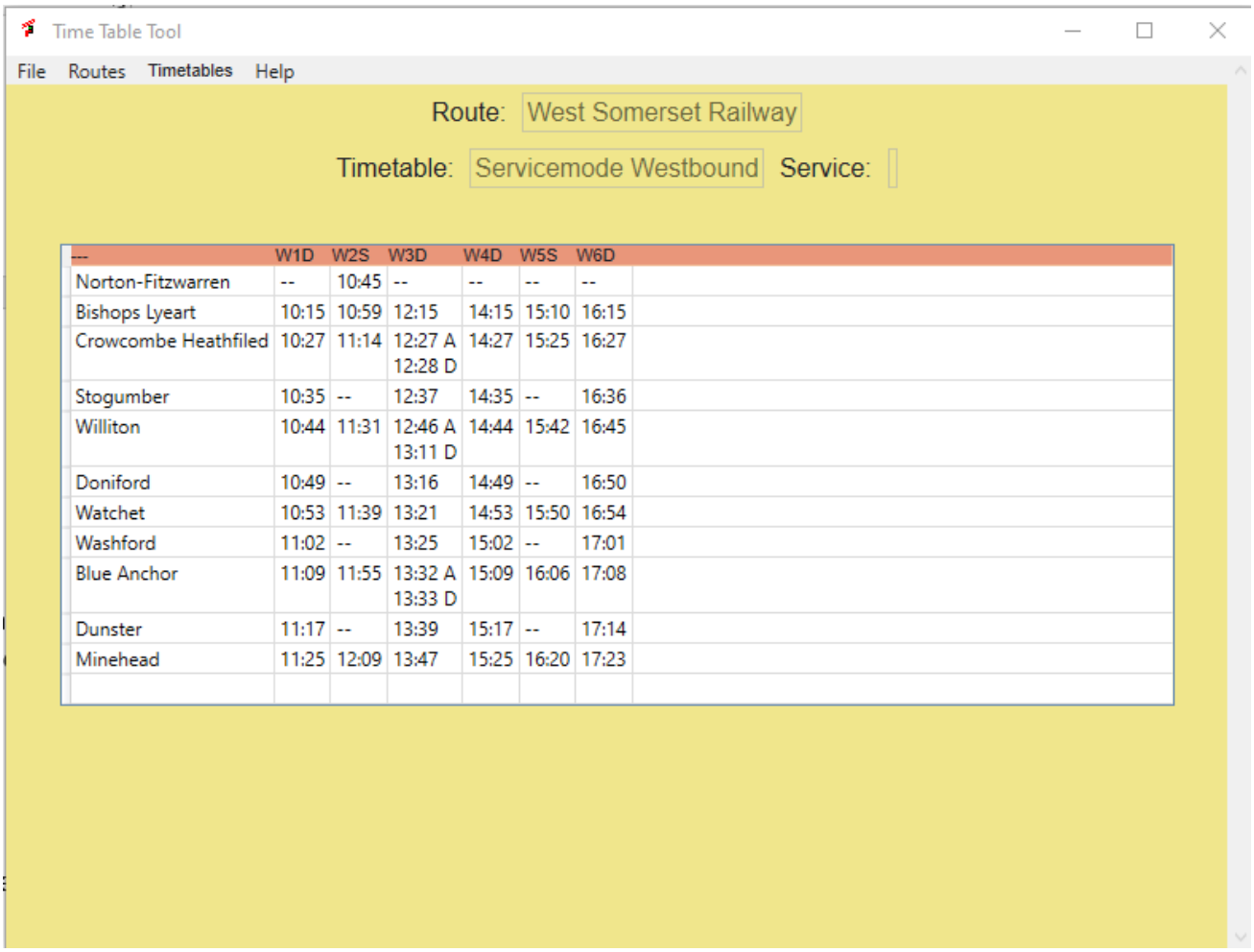
Finally, you may want to delete all backups and make new backup.

You can postpone this step, but the Delete functions will be disabled, because the old database does not support deletion of data in the database. This is a bit unfortunate, but this workaround is OK.

For details on backup and restore, see chapter 6.7. For details on export and import see chapter 6.8

4 About timetables

At the moment, TimetableTool will create two different views on a timetable. The first one is the classic timetable, as you can get them in the well-known timetable booklets. You need to create one timetable for each direction and these will not be combined. An example is shown in Figure 1 Classic timetable view as generated by TimetableTool Figure 1.



---	W1D	W2S	W3D	W4D	W5S	W6D
Norton-Fitzwarren	--	10:45	--	--	--	--
Bishops Lyeart	10:15	10:59	12:15	14:15	15:10	16:15
Crowcombe Heathfiled	10:27	11:14	12:27 A 12:28 D	14:27	15:25	16:27
Stogumber	10:35	--	12:37	14:35	--	16:36
Williton	10:44	11:31	12:46 A 13:11 D	14:44	15:42	16:45
Doniford	10:49	--	13:16	14:49	--	16:50
Watchet	10:53	11:39	13:21	14:53	15:50	16:54
Washford	11:02	--	13:25	15:02	--	17:01
Blue Anchor	11:09	11:55	13:32 A 13:33 D	15:09	16:06	17:08
Dunster	11:17	--	13:39	15:17	--	17:14
Minehead	11:25	12:09	13:47	15:25	16:20	17:23

Figure 1 Classic timetable view as generated by TimetableTool

You can export this as a.csv file to process it further outside the tool. You are completely free to represent a number of additional locations. As you can see, if a train stops longer than one minute, separate arrival and departure times can be included. If a train should not stop, automatically two dashes are included to represent this situation.

There is second view, representing the timetable in a graphical way. In this representation you normally include both directions, but it is possible to create a timetable that shows just a single drive or all trains in one direction. It is up to you. This is a really powerful tool, which is also used by real world time table designers.

- It shows the density of trains at a track.
- It shows where trains meet, so how many trains do you actually see?
- It shows conflicts on single track areas

An example is shown in Figure 3.

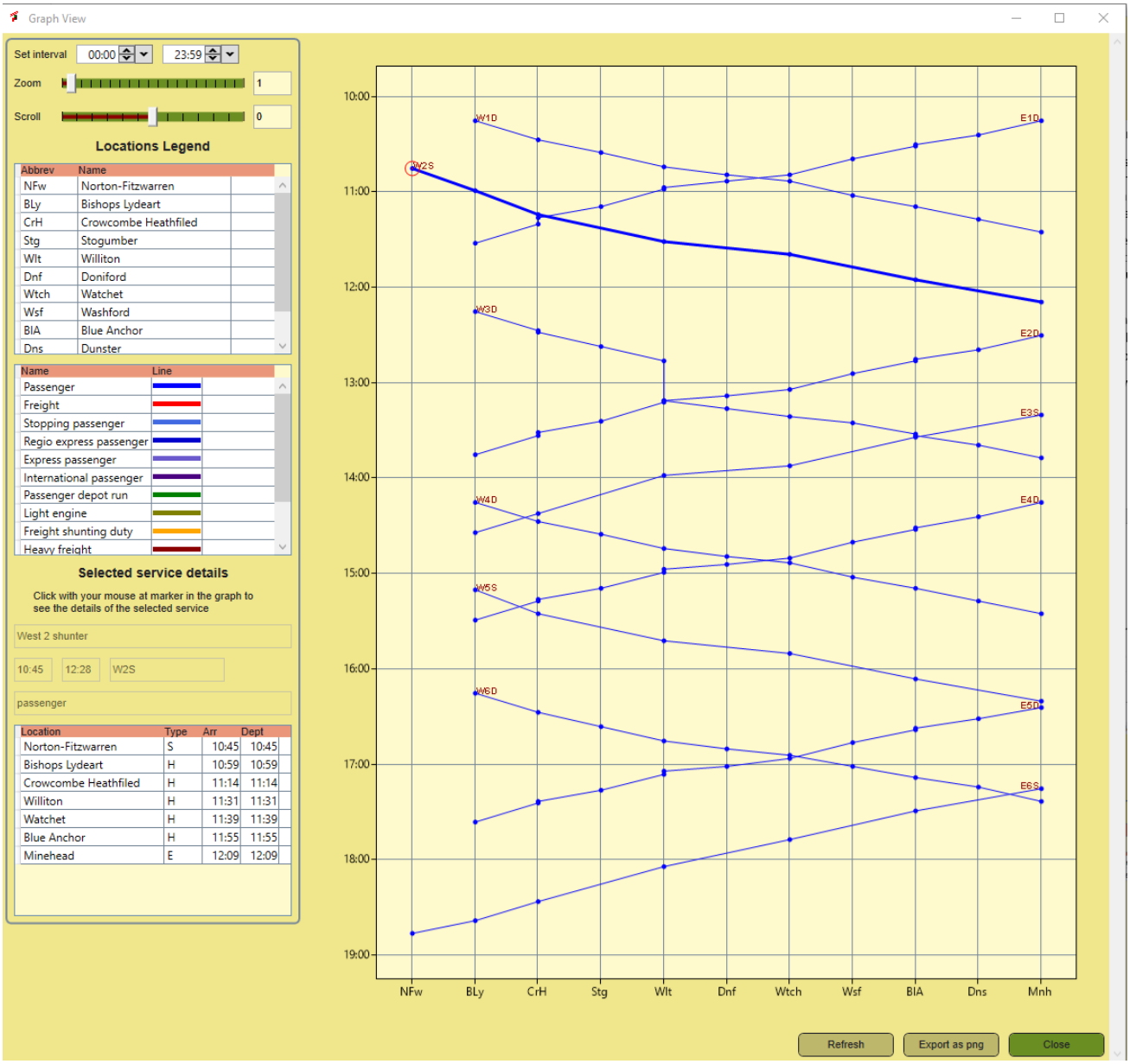


Figure 2 Example time graph for West Somerset Railway.

The red squares denote the starting points. At the bottom you see abbreviations for stopping locations. If you click with the left mouse button at one of the blue points, this will retrieve the service details. These are shown at the left side of the screen. If there are many services, you can use the zoom and scroll slider to enlarge the time scale.

Please note, this is an alpha product. I am very proud of the results until now, but I want to make it more interactive and new representations of timetables will be created. If you have specific wishes, let me know.

Some ideas:

- An Ebula like view, which gives specific information to the driver about speed limits, gradients signals and so on.
- Views that show how rolling stock is moving over the day.
- Views adding crew plans

I only tested this using TSW routes, but you can use it for other train simulators as well and possibly also for bus simulators or other simulation games that require scheduling.

It is NOT intended for real world timetable development. I do not have the professional skill to develop something for that.

A third representation is the arrival/departure timetable for a specific location or station. In the example, you see these for Williton, but you can select other locations as well.

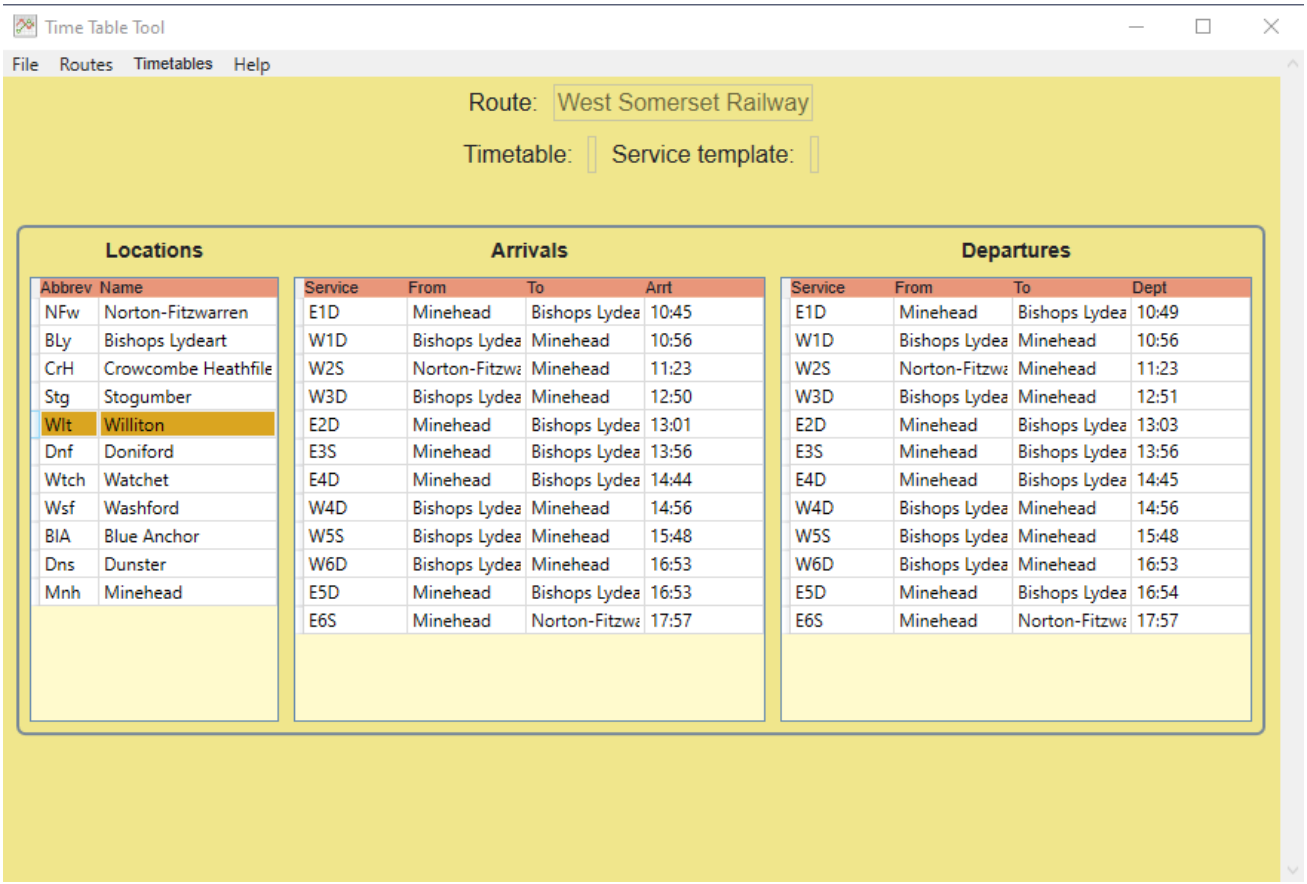


Figure 4 Arrivals and departures for a selected location

5 Creating a Timetable

TimetableTool is a design tool for timetables and works mostly bottom up. I have specifically timetable mode for TrainSimWorld in mind. This requires the ability to repeat a drive several times on a day. This way of working is supported by TimetableTool. At the moment, it requires al little bit additional work, but I will someday provide some more shortcuts to make it go faster.

For now, you need to perform the following steps:

1. Define a **route**. All timetables are bound to a route.
2. Add an number of named **locations** to the route. A location can be a station, a siding or any other point of interest. You also must define an **order** for the locations.
3. Define **directions**. For instance, you may drive from west to east or the other way round. The tool must be able to distinguish between the different directions. In most cases two directions should do.
4. Define **service templates**. A service template is a container to hold stops, driving times but not a fixed departure time. This mak s it possible to run a service multiple times. Defining services is to most work, so I included some optimizations.
5. A **service** makes the service template a real drive, by giving a service templates one or more departure times. In the graphs you see services.
6. Bundle a number of services in a **timetable**. A timetable is what you finally want to see. You can have as many time tables as you like! It is nothing more than a specific view on the data.

You need to execute these steps in the order shown above. TimetableTool will force you to do so, by disabling menu items that are missing prerequisites. In the rest of this chapter, I will clearly indicate these requirements.

This chapter has a tutorial nature. In the reference section will cover a lot more details on how it works what you can do.

5.1 Define a route

In Figure 5 you see the screen as it looks like when you first open TimetableTool. For this tutorial, we will create a small new timetable for the Ruhr-Sieg-North route.

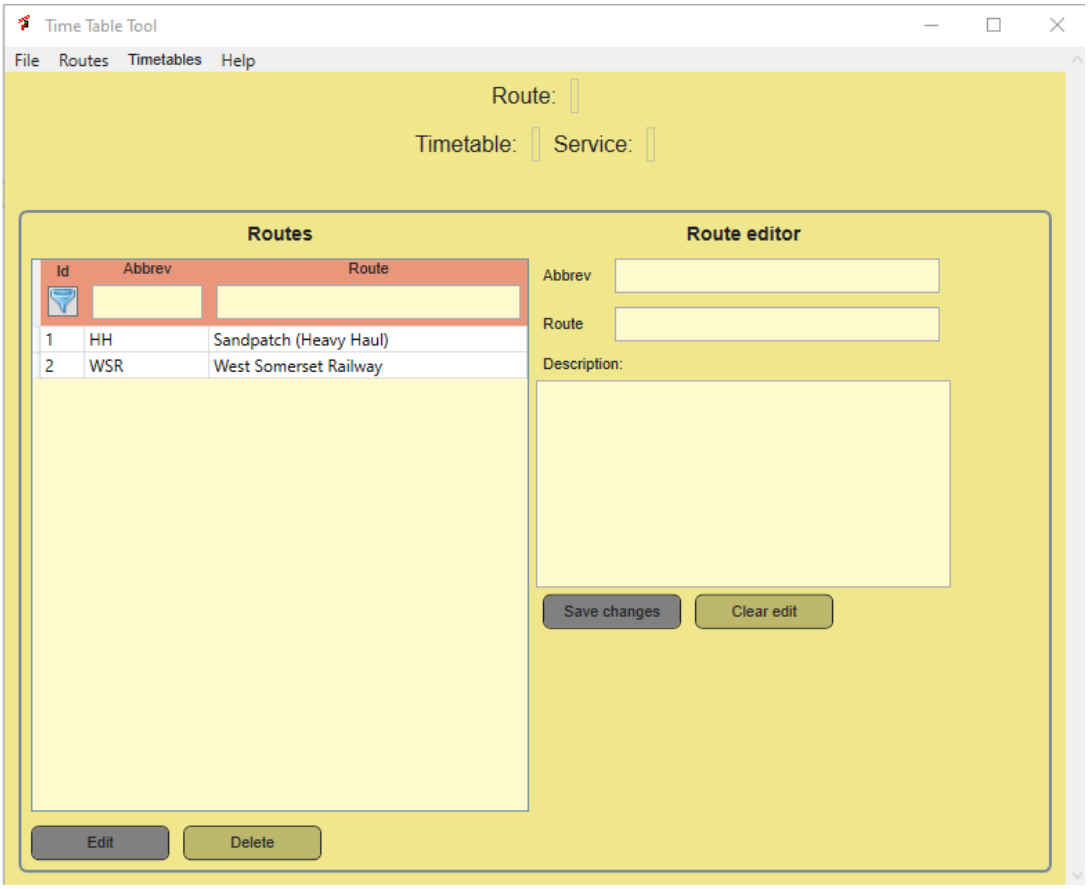


Figure 5 Route selection and entry screen.

At the top of the screen, you can see the name of the currently selected route, service and timetable. As you see, nothing is selected and in the left pane you see the routes for the demo timetable. We can ignore this for now.

To enter a new route, complete the form at the right. All three fields are required. Once you have done, the Save changes button is enabled and you can save the route into the database. It will appear right away in the route list in the left panel.

Now, you need to select the route you just created, to enable the locations screen.

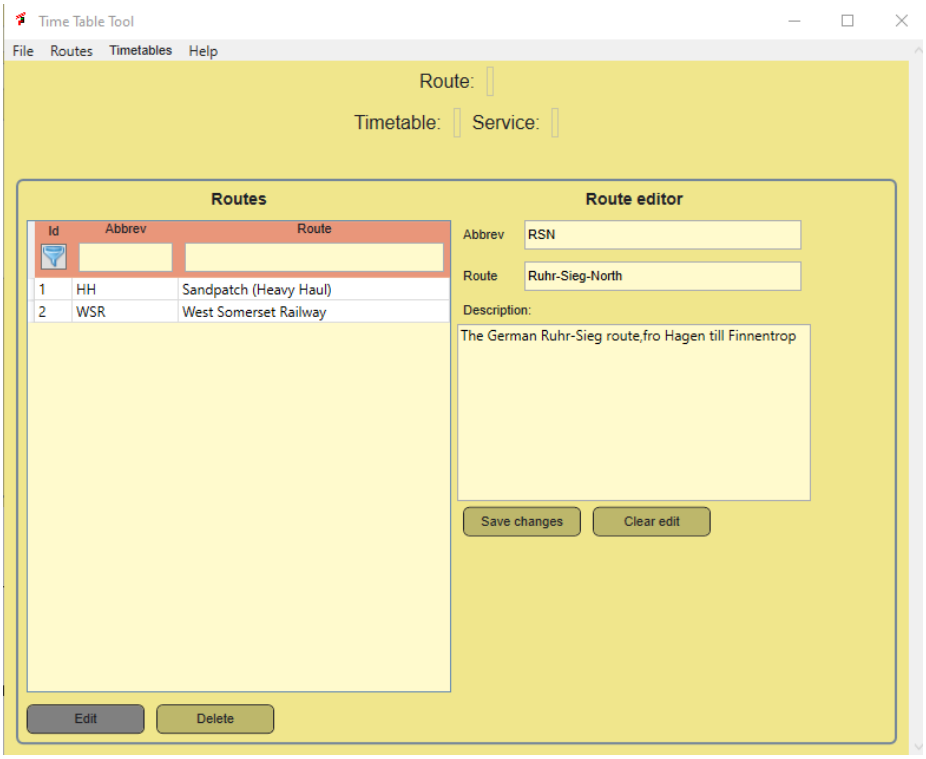


Figure 6 Completed Route edit form

5.2 Locations

If you selected a route, at the top level, in the Routes menu the menu-item Location will be enabled. In the route table you see the orange background, indicating the route is selected and the name of the selected route is shown in the text above your screen.

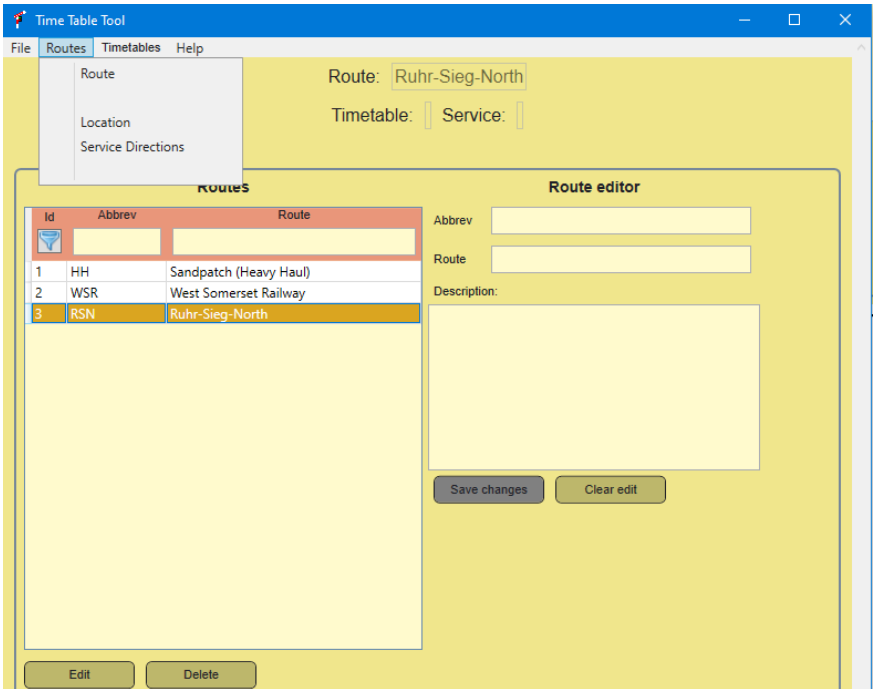


Figure 7 A route is selected, which enables some other screens, e.g. locations

No you can open the locations screen, using the menu. The locations shown are filtered by route. As you just created the route, there are no locations yet, so the table is empty.

Because we use a database, and you can add locations afterwards, you need to tell TimetableTool in which order trains may pass the locations. You do this by assigning each location a number, called Order. Locations will be sorted according to this number. The value is not critical, it must be a number, that is all. I recommend to increase location numbers by 10.

In the reference guide I will explain how to work with branches, like in the Rhine-Ruhr Osten route. For now we keep it simple.

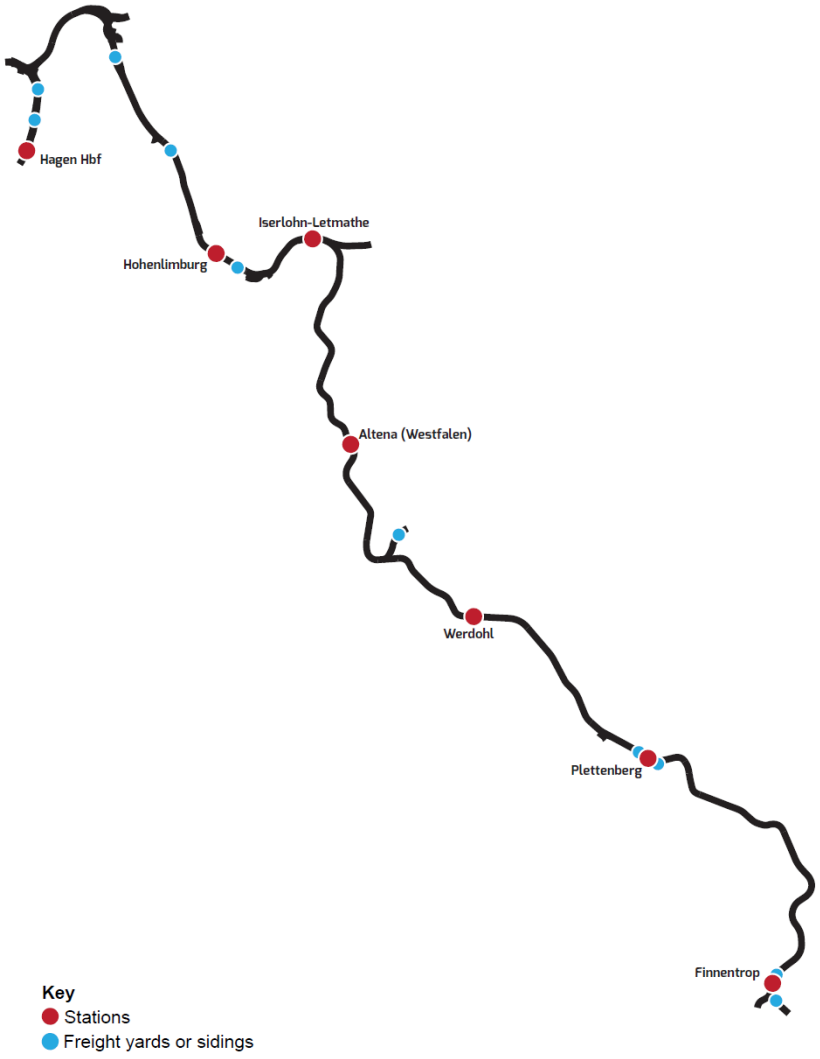


Figure 8 Route layout as shown in the official game manual

In Figure 8 you see the route as presented in the game manual. Freight yard did not get names. This is not really bad. You can change names for locations later, if they are incorrect, without having serious impact on the timetables.

I decided to include the line ends as locations. AI trains may depart from Finnentrop to the line end. I also included the freight yards, because we will need them to add freight services. It does not matter where you start, I decided to start at Hagen and then number southward to Finnentrop.

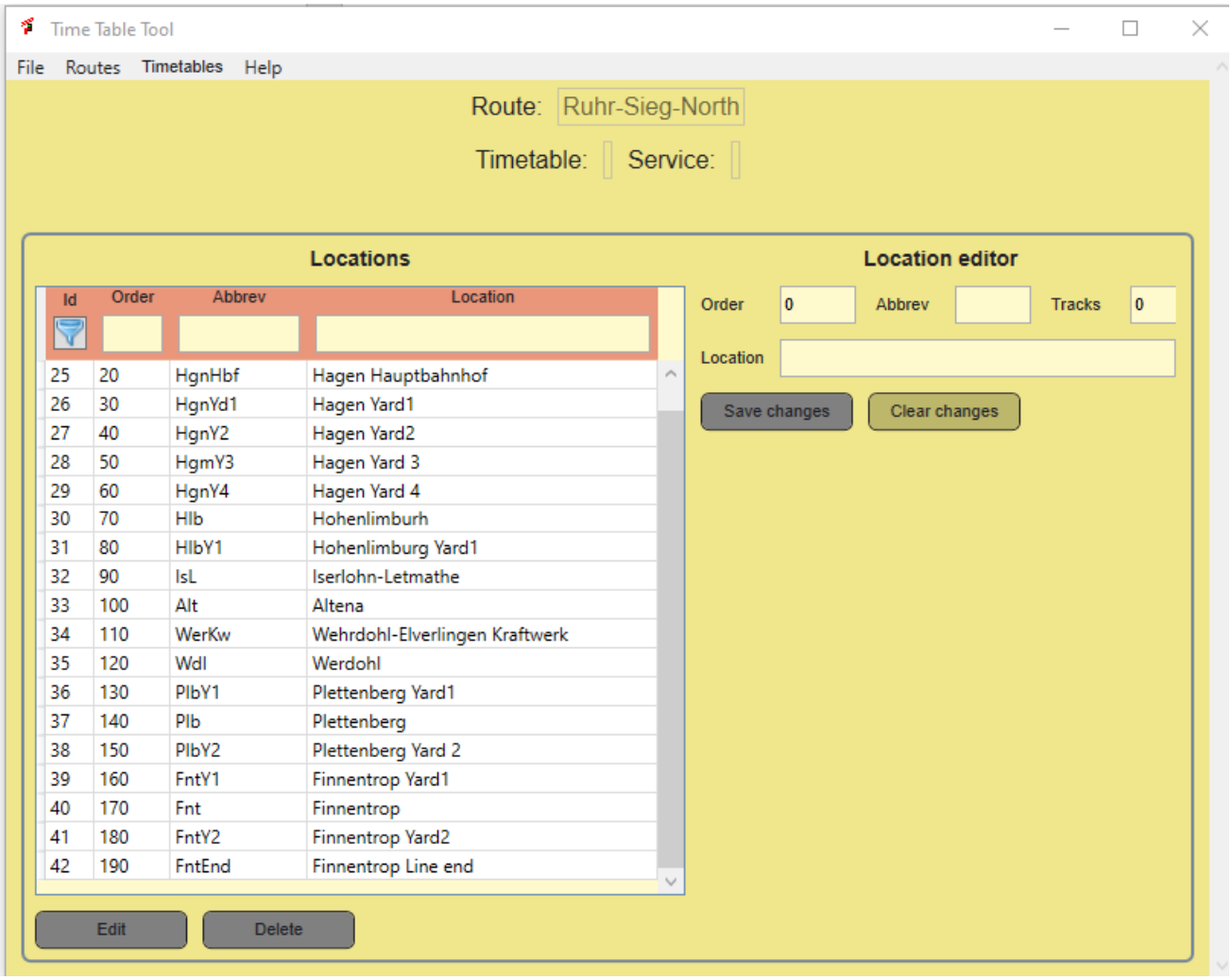


Figure 9 Completed location list for the RSN route.

The completed list may look like Figure 9, depending upon your choices. I used fantasy abbreviations, but if you know the official abbreviation, you can use them as well. There are no restrictions, but I recommend to keep them short, less than 6 characters. You also must fill the number of tracks. The basic idea is that this can be used to have an idea about the capacity of the location. This is especially interesting for single track routes. I am not yet sure if I want to keep this information.

I increase the order by 10, to make it easier to add a location later.

There is no need to select one of the locations. You can now proceed to set up the service directions, which is straight forward.

5.3 Service directions

Service directions refer to a route. So you must have selected a route in order to access this form. You can open the form through the Routes menu

It is straight forward, create a name and an abbreviation. These are only used for reference. Internally the database identification is all TimetableTool needs and you need to tell if the direction should follow the order of the locations from low to high or from high to low (descending order).

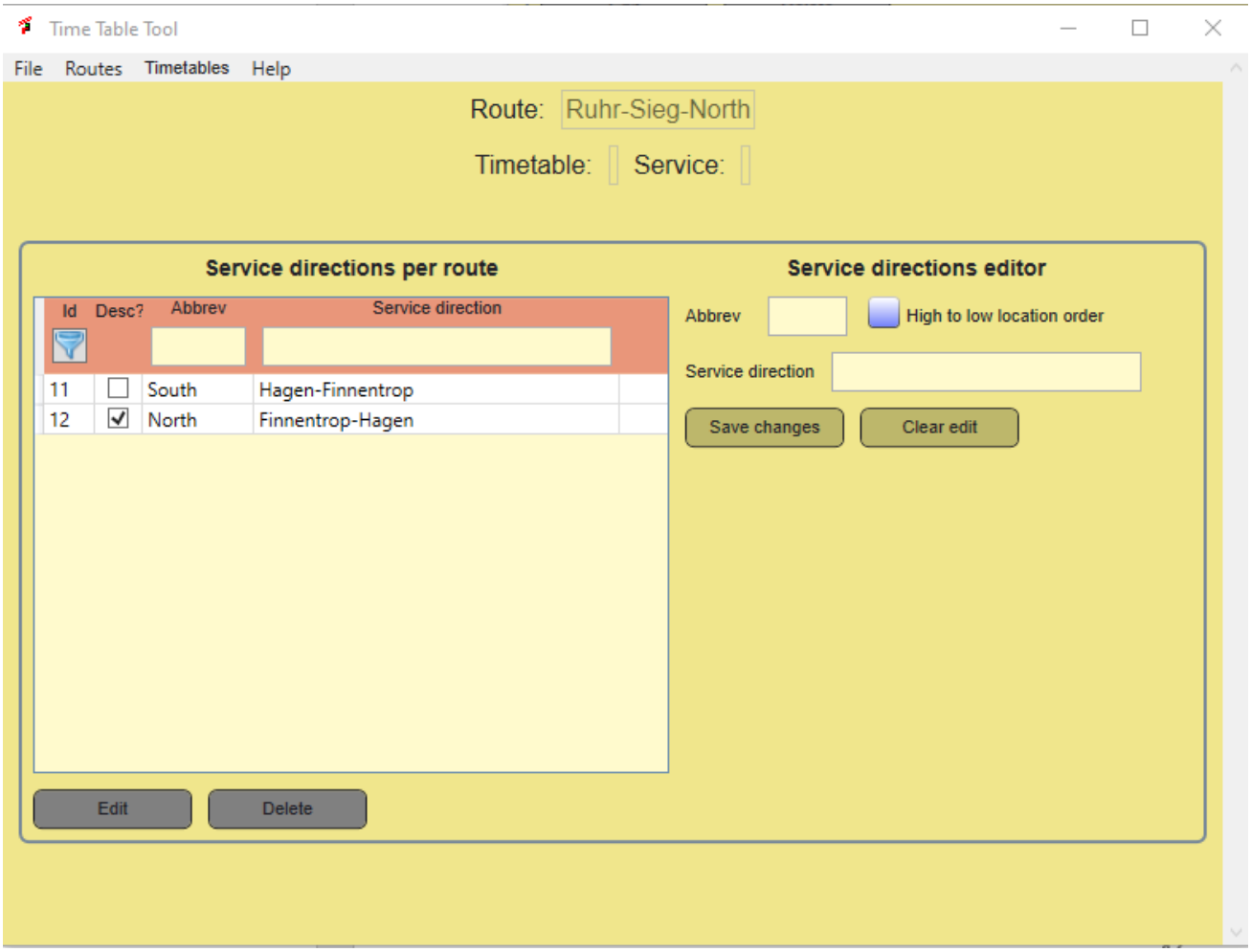


Figure 10 Service direction table filled for RSN

Normally two directions are sufficient. As you see, in the abbreviation I use the global compass direction, which is described more precise in the name. The “Desc” field or High to Low Location numbers tells TimetableTool to start with high location numbers for trains from Finnentrop to Hagen.

Once this is done, the route details are set. We now can define services.

5.4 Service templates

In order to add service templates, you must have selected a route. Then, at the Timetable menu you can click at the service templates menu item. Do NOT confuse this with the services menu item!. The service templates page allows you to define a set of re-useable templates. These will be used to define specific services later. A service template defines all stops and drive times, but it does NOT set a start time. So, later you can repeat the same service template a number of times..

For RSN I used the timetable you can download here as basis:

https://www.railsim-fr.com/forum/index.php?/files/file/1449-tsw-fiches-horaires-ruhr-sieg-nord/

We will create 3 service templates in each direction, but note they can be re-used.

		RB91	RE16	RB91	RE16	RE16	RB91	RB91	RB56	R
Hagen	Départ	05:40	06:10	06:40	07:15	07:40	08:15	08:40	09:06	0
Hohenlimburg	Départ	05:52	--:--	06:52	--:--	--:--	08:27	08:52	09:18	-
Letmathe	Départ	05:57	06:26	06:57	07:31	07:56	08:32	08:57	09:23	0
Altena	Départ	06:06	--:--	07:06	--:--	--:--	08:41	09:06		-
Werdohl	Départ	06:16	06:43	07:16	07:48	08:13	08:51	09:16		0
Plettenberg	Départ	06:24	06:51	07:24	07:56	08:21	08:59	09:24		0
Finnentrop	Arrivée	06:34	07:01	07:34	08:06	08:31	09:09	09:34		1

Figure 12 Part of the North-South passenger timetable for RSN

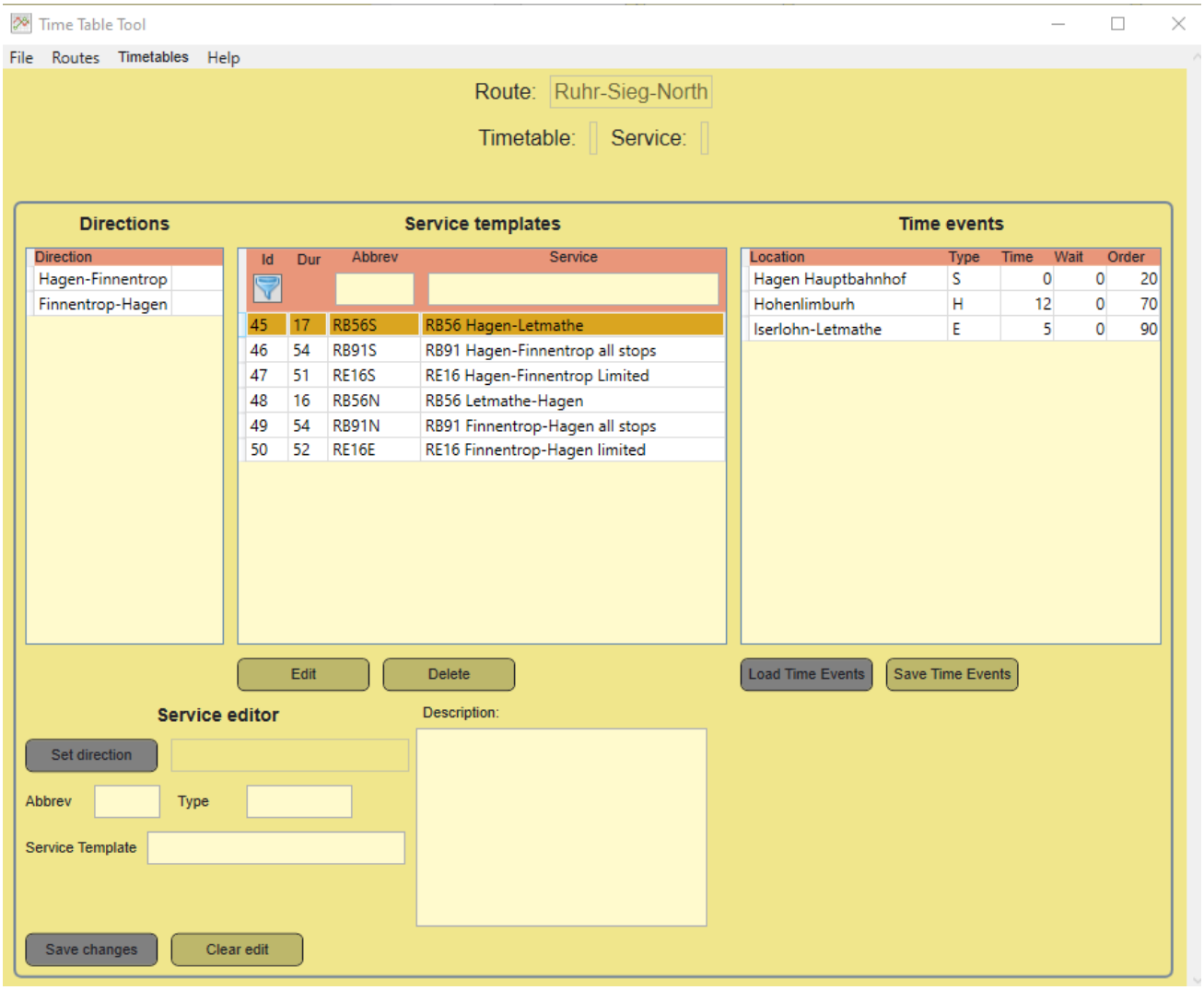


Figure 11 Service templates form

When you have look at this timetable, You see there actually are just three different services: RB91, RE16 and RB56. The others are just copies. This step is by far the most complicated procedure. TimetableTool has a shortcut here to make it easier to create these service templates, as you will see soon.

As you see, the window is divided in three tables at the upper part and has an edit window at the lower part. At the left, you see the directions that are available. You must tie each service to a direction, which is done by selecting a direction and then add it to the service, using the Set Direction button in the Service editor.

The other fields are trivial. You select an abbreviation (e.g. RB91), a descriptive name and there is room for a more extensive description. All fields are mandatory. If they are all set, you can save the new service, using the Save changes button.

Now you can select the service. This step will fill the third column with all locations. You can directly edit this table, which makes it a lot faster that the old way. The old way is still available, see the Reference Guide chapter for details.

I will demonstrate the RB56 service first, because it is very simple.

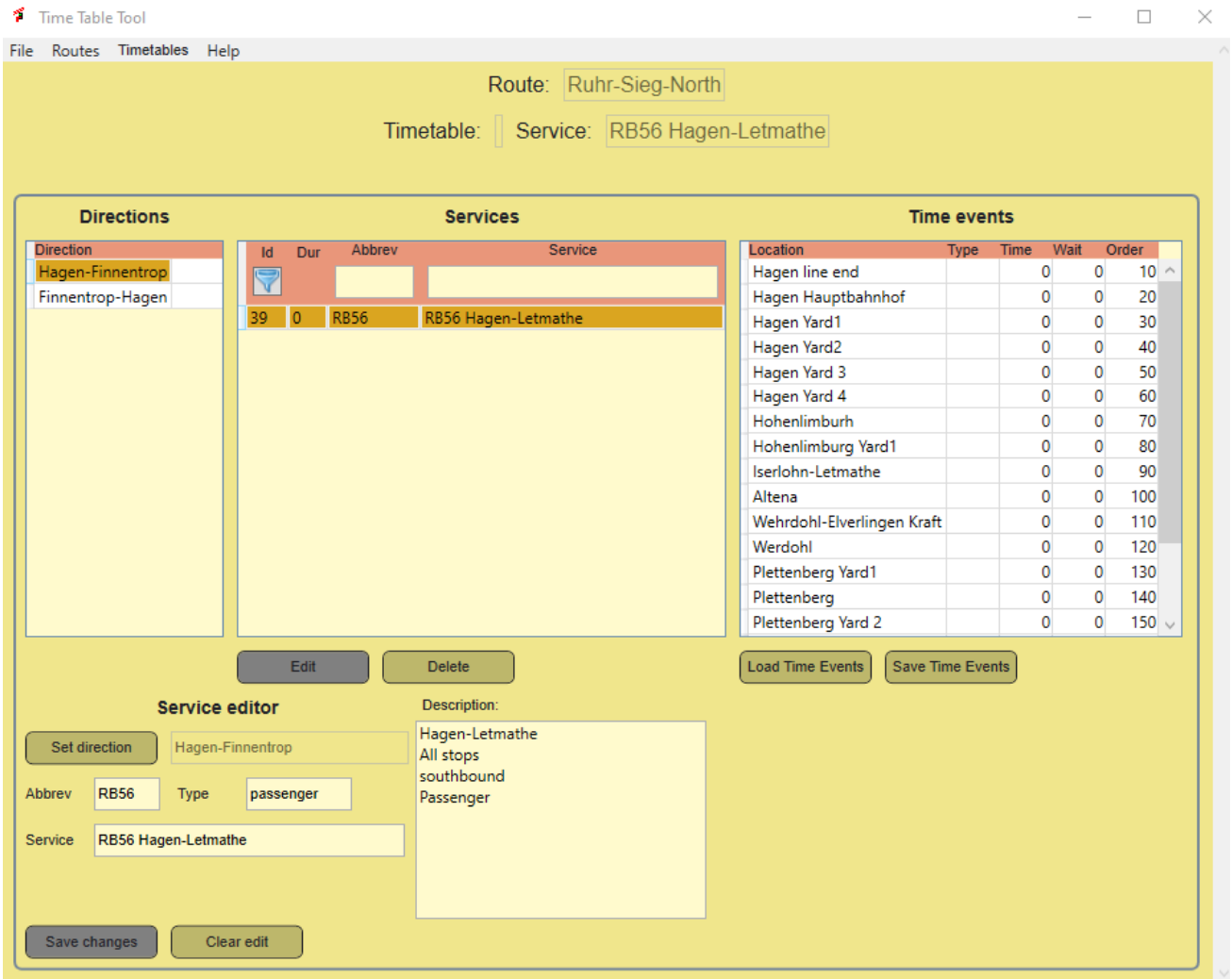


Figure 13 RB56 service defined and time events loaded

In Figure 13 you see the results of the first steps. A service template is created and saved. Then the service is selected and prepared to edit (Edit button). Once you have done that, the button Load Time Events is enabled. If you press that button, it will load either the previously defined time events or load all locations, so you can create time events. The filled type must be selected from a predefined list.

If you need to use a service type that is not supported, contact me, and I will see what I can do. I decided to make it not configurable by users directly, because that would cause a lot of issues in future.

Note: you may notice an older version of the form is shown here. This should not have impact on the way of working.

This is great. We must have a closer look at the time events table. This is where most work is needed and where it happens:

Field name	Description
Location	You see the location name here, but behind it are all details in the location. Locations are sorted by order and take into account the service direction.
Type	The type of the event. If you leave it empty, the location will NOT be saved as Time Event. I use following codes: ■ S=start, where the service starts ■ E= end, where the service terminates ■ H= halt passenger stop ■ P= pass, pass through with a set time ■ -- = keep the Time Event at this location, but it is not scheduled. ■ SE= train starts and ends in the same location, e.g. local shunter.
Time	The relative arrival time from the last scheduled location till this location in minutes. For the first location, it always should be 0.
Wait	The waiting time at the location. This is optional. If it is zero, TimetableTool assumes arrival time is equal to departure time. If a value is set, separate arrival and departure times will be generated (as you can see in the WSR timetable).
Order	This is NOT the order of the locations, but the order of the Time Events. It is generated automatically, but you can change it and changes will be preserved. The nice thing is that it takes the service direction into account. If you have set to descending checkbox, it will reverse all.

Time event types are used to create the arrival/departure tables. Make sure to use the S and E Event Type as indicated above. In later versions I will and some checks to make this easier.

You now can edit the time events directly in the table. For the service template we create, the result should look like this:

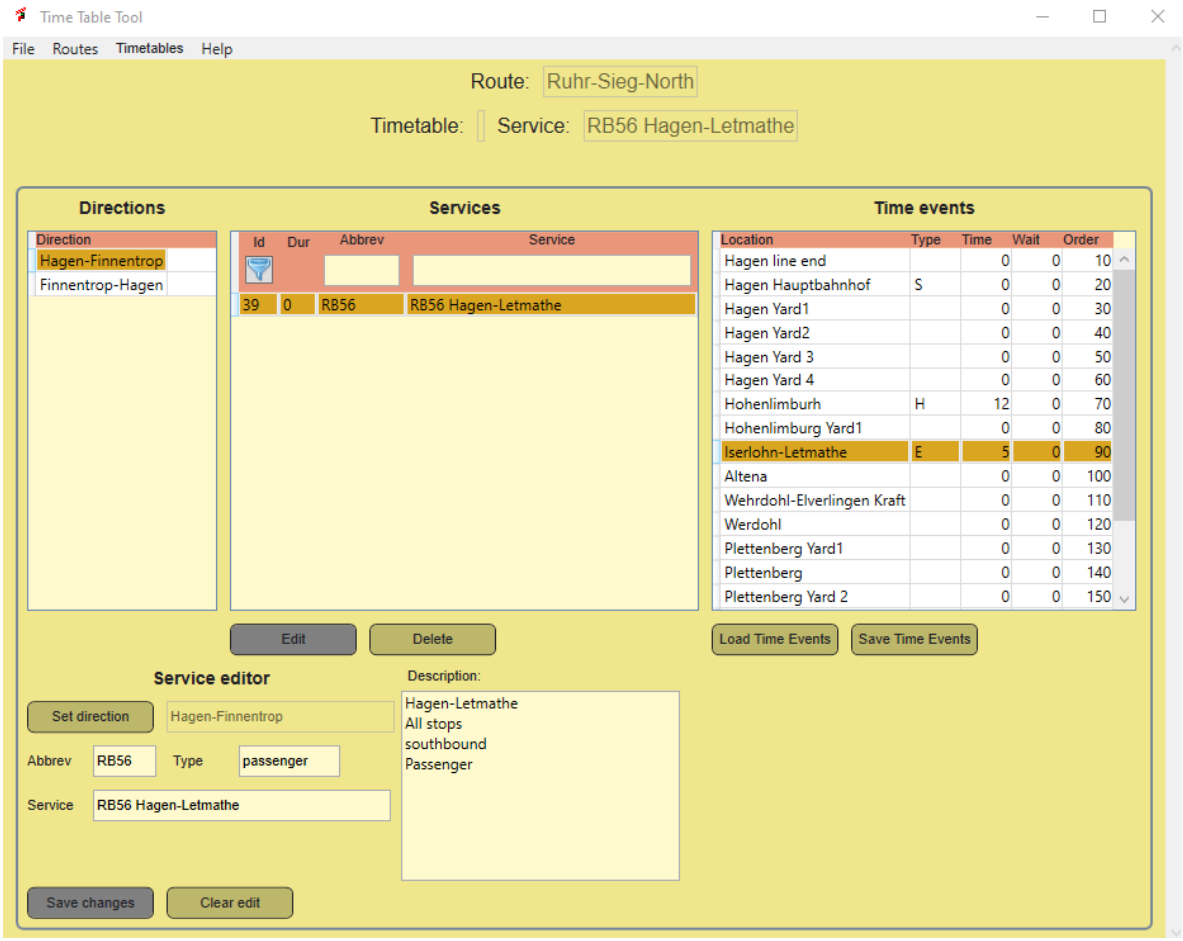


Figure 14 Completed time events for service RB56

You see, I set Hagen Hbf as starting point, giving the Type the value S.

Then, you need to calculate that it takes 12 minutes to drive to Hohenlimburg. This location gets H for Type and 12 for time.

Finally, it takes 5 more minutes to drive to Letmathe, where the service ends, so I make that clear by putting S in the Type.

Now you should save the Time Events. Resulting in this screen:

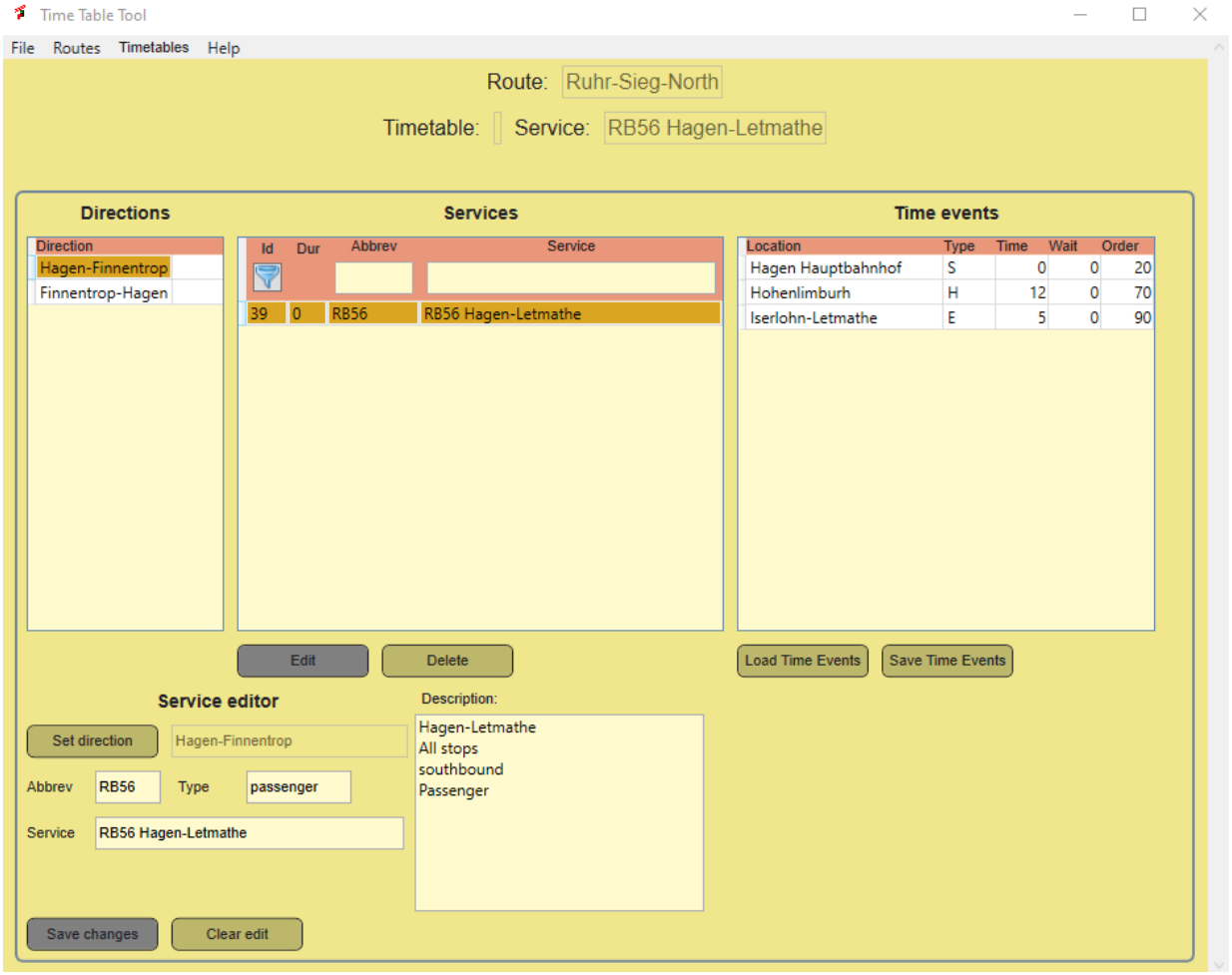


Figure 15 Saved time events

You see now that all time events we no longer needed are kicked out again. The service should show the total driving time, but this is at the moment of writing not working well. It is calculated but not shown right away.

	Service type and Time Event Type are not used at the moment. It is possible that they will get more clear definitions. I recommend to use these fields as described here.

I the same way I will now create the other two services and show just the time events:

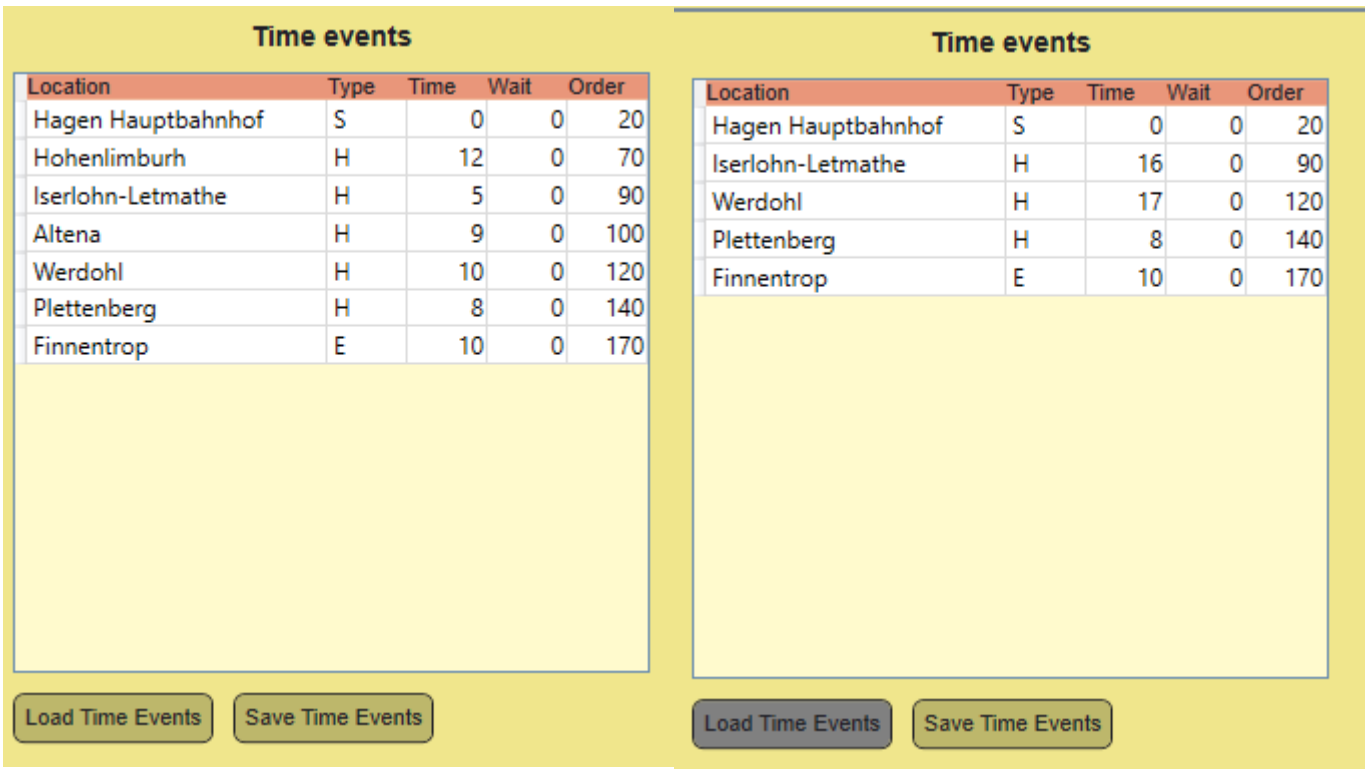
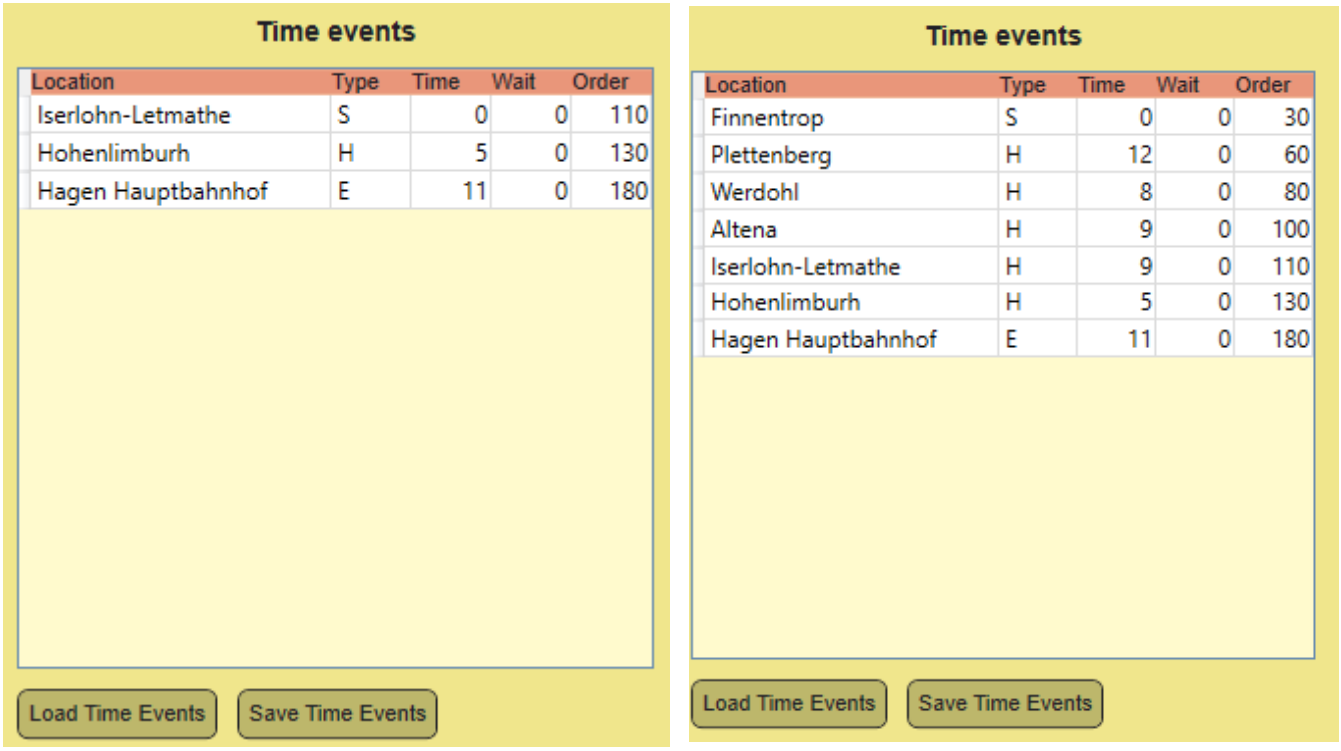


Figure 16 RB91 en RE16 services

For the total demo, I will add three service templates for the other direction as well. I see now, DTG has used the same service names. That is not a very good idea, there I changed the abbreviation to include the direction.

So, RB91 will be RB91S and RB91N.



5.5 Services

Services templates do not have a start time and they may be repeated several times a day. In this step, based on a service template, one or more services will be defined. In the RSN example you see that a number of services are repeated. These can share a single service template and we will create services based on a single service template to repeat similar drives.

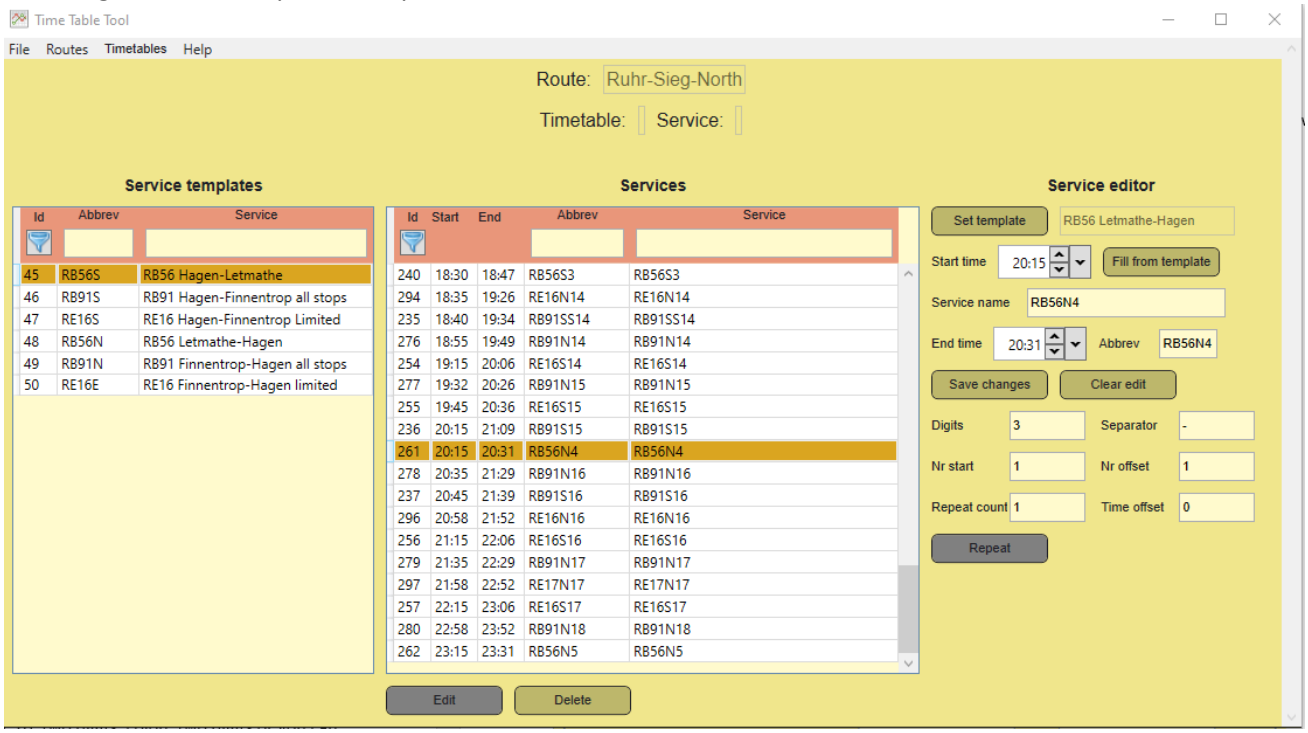


Figure 18 Define services

In order to define services, the route must be selected. Once you select Services from the menu, the window should look like Figure 18. It has three columns:

The left column shows all services you defined for the route. The middle column shows the services and the right column is the editor you can use to define a service.

To create a new service, proceed as follows:

1. Select a service template in the left column. In the right column, choose this service template with the Set template button.
2. Set the start time, you can simply type it in the format 11:20, two digits, colon, two digits or you can use the up/down buttons. I do not like these buttons, someday they will be replaced. Tick the down arrow at the right of the Start Time filed briefly. (Sorry, this is annoying. I use an external component that is designed for a different .Net version and it does not do exactly what I want).
3. Optionally, click the Fill from template button to pick some data from the template.
4. Give the Service a name
5. Give the Abbreviation a name
6. Set the end time (this is not yet used), or correct it.
7. Once all fields are completed, press Save Changes.

For the abbreviation, you may use a train number or something like that. In this case, I just use the service abbreviation followed by a sequence number to make the name unique. The name field is not essential, you can fill it with a more descriptive text if you need it. I just use the abbreviation here. The abbreviation is shown in the displayed timetables, but the name is used as a tooltip.

Since the timetable is not very long, I think I can add all services. You must be a bit careful, sometimes the services look similar, but the timing may be different. I check the total duration, if that matches, it is a real re-use.

I completed this by adding all services to the service list. It still is quite a bit of work, but in version 0.2.0 there are some shortcuts. See chapter 6.2.6 for a detailed explanation, this is above the level of a tutorial.

5.6 Timetables

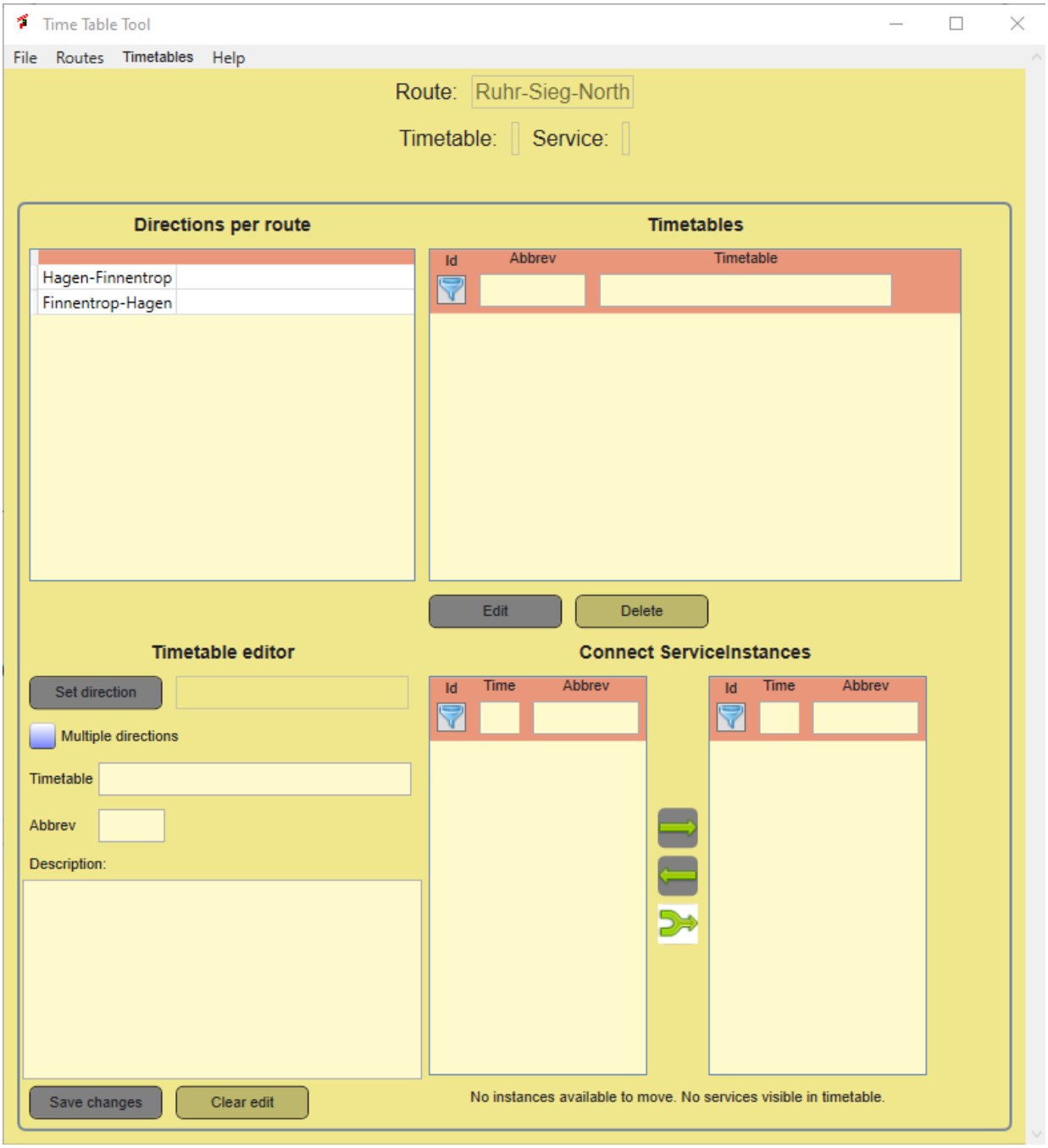


Figure 19 Timetable window

The last step is to add one or more timetables. There are a few restrictions. Essentially a timetable is just a set of services you like to present somehow. You can make as many as you like, but I recommend to make at least three of them:

- One for each direction, so at least two
- One that combines all direction.

In order to create a timetable you need to select a route. Then you can open the timetable window, which initially looks like Figure 19

It may look a bit complex at first sight, because it essentially combines four parts:

1. At the upper left, you see the service directions that are available. You may need to select a service direction, in order to create a timetable that is bound to a specific direction.
2. The upper right part shows all timetables you defined.
3. The lower left is used to set up a new time table or to edit an existing one.
4. The lower right corner is used to bind service instances to your timetable.

First, let us create the southbound timetable. You do this like this:

1. Select a service direction in the upper left part.
2. Click the Set Direction button
3. Give your timetable a name and an abbreviation, add a description.
4. Now, you can save the timetable
5. Select the new timetable and now you see in the lower right corner all available services, that match with the selected service direction.

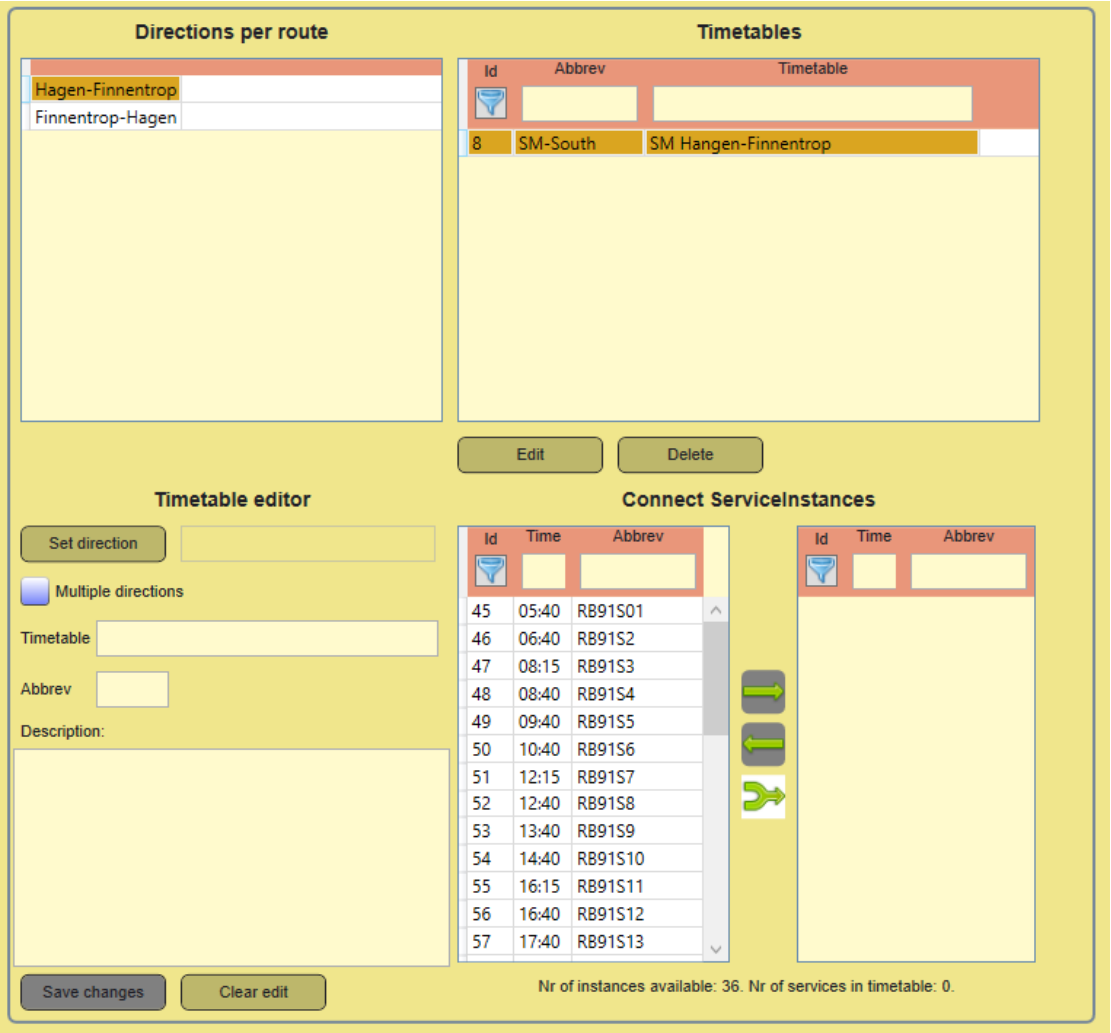


Figure 20 Selected timetable showing available services

Now you have two options:

- Select a service in the left column. Press the green right arrow button and the service instance is added to the timetable.
- You also can press the lower forked arrow, and add all services to the timetable at once.

If you want, you can remove a service, by selecting it at the right side and click the left arrow. You can repeat this as often as you like.

You may notice that only services that match the service direction are selected.

You do NOT need to save your work explicitly. Each change is saved right away into the database.

For the third timetable kind, you should NOT select a service direction, but you click the checkbox “Multiple directions”. In this case all services can be added to the timetable, but for this timetable you cannot choose the table form of the timetable. You are done now with the preparations.

5.7 Show the timetables

In the file menu you can select the **Reports** item. This will show a screen, with all timetables:

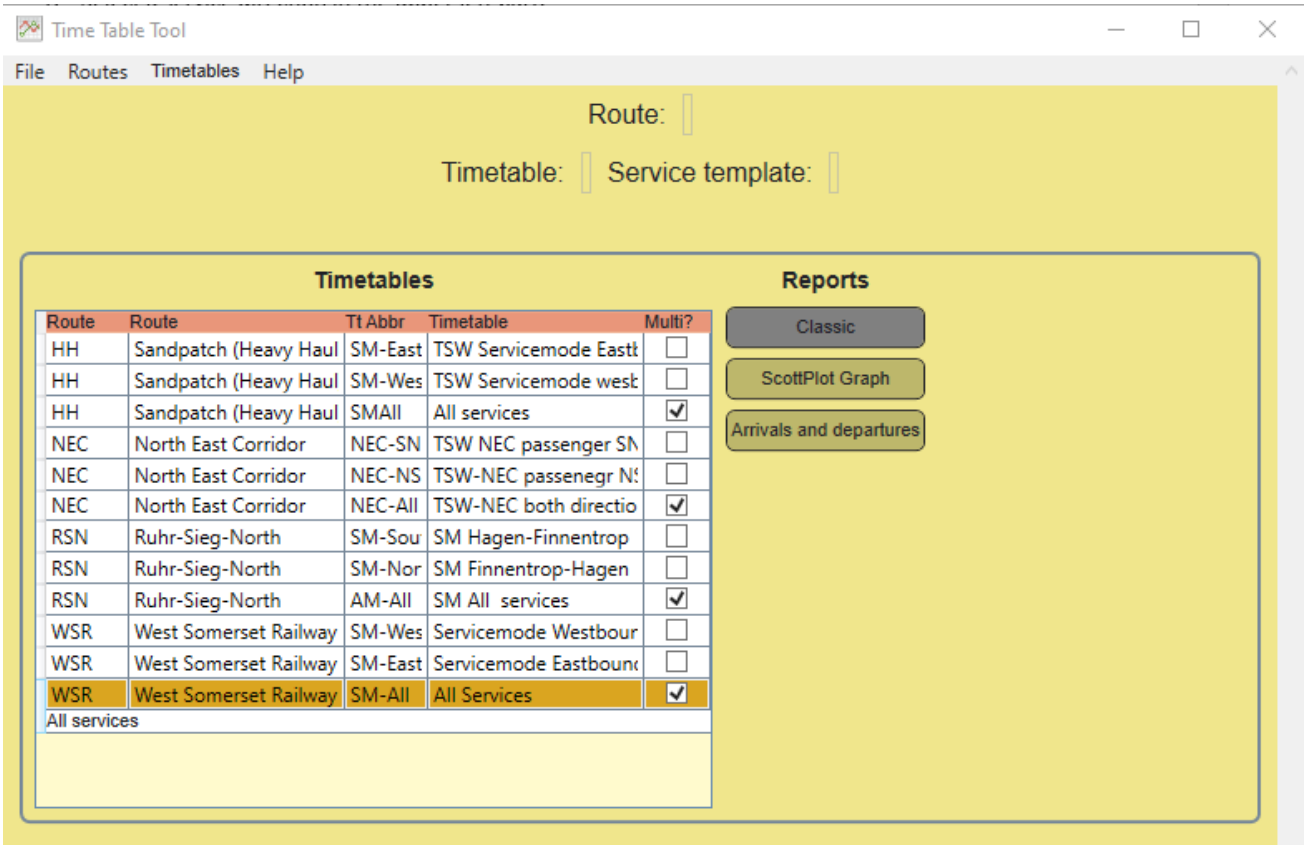


Figure 21 Reports screen

In the left column you see a list with all defined timetables and the routes to which they belong. You can select the timetable to display here. There is one restriction:

You cannot show a timetable with more than one direction in the classic format. This will stay disabled if you select a timetable with the Column “Multi?” checked. Once you selected a timetable, press the buttons at the right to view the time table. In future more options will be added here.

6 Reference guide

I assume you read the tutorial, so stuff covered there will not be repeated here. However some advanced topics are explained in this part of the manual.

6.1 Locations

For locations, you need to tell explicitly how you order locations. How can you do that for routes like Rhein-Ruhr Osten, which has two branches or with lines that have branch lines? It is quite simple, let show that in an example.

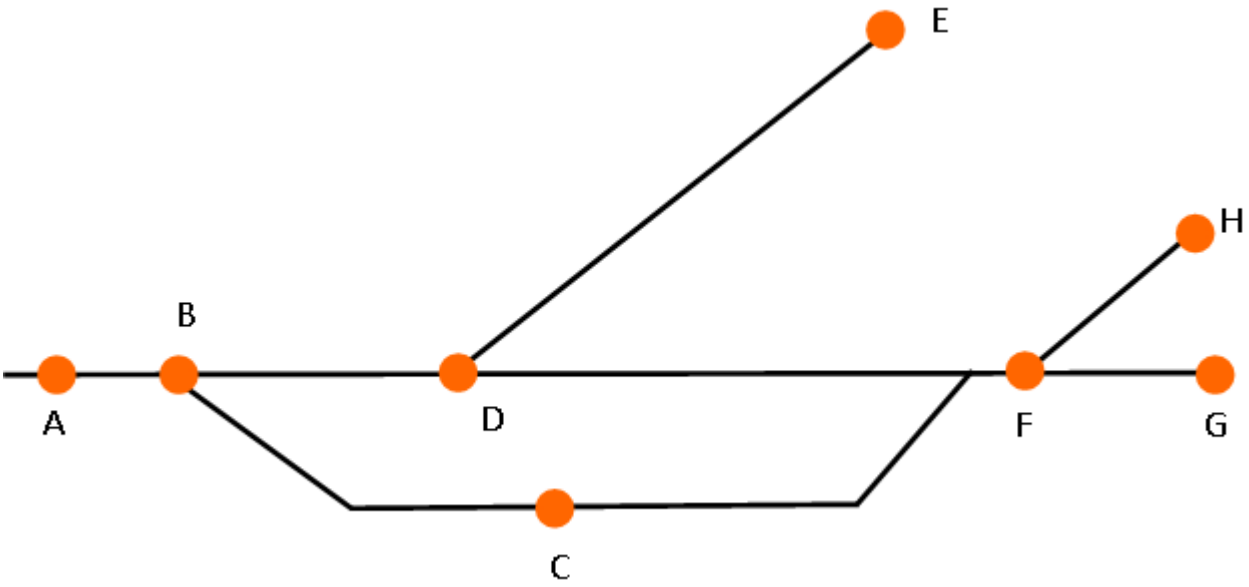


Figure 22 Example of a complex network

In the example as shown in Figure 22 you see a fairly complex route network. You can drive for instance starting at A then go via B, C and F to G. The only requirement for the location order, is that each location, you can pass, seen from left to right has a higher number.

For C and D, there are no requirements, you cannot go from C to D, so no ordering is needed. F must have a higher number that both C and D, because both C and D are connected to F.

If you comply with these simple rules it will work. So, you may do it like this:

Location	Order
A	10
B	20
C	30
D	40
E	50
F	60
G	70
H	80

One way to do it, make a diagram and put a ruler on it. If you number strictly from left to right, you will be fine.

Notice I use intervals of 10. This makes it easy to add locations later.

6.2 User interface

In this section, some information on the user interface is given in a structured way. It may be helpful to understand a bit on how it is designed.

6.2.1 Menu

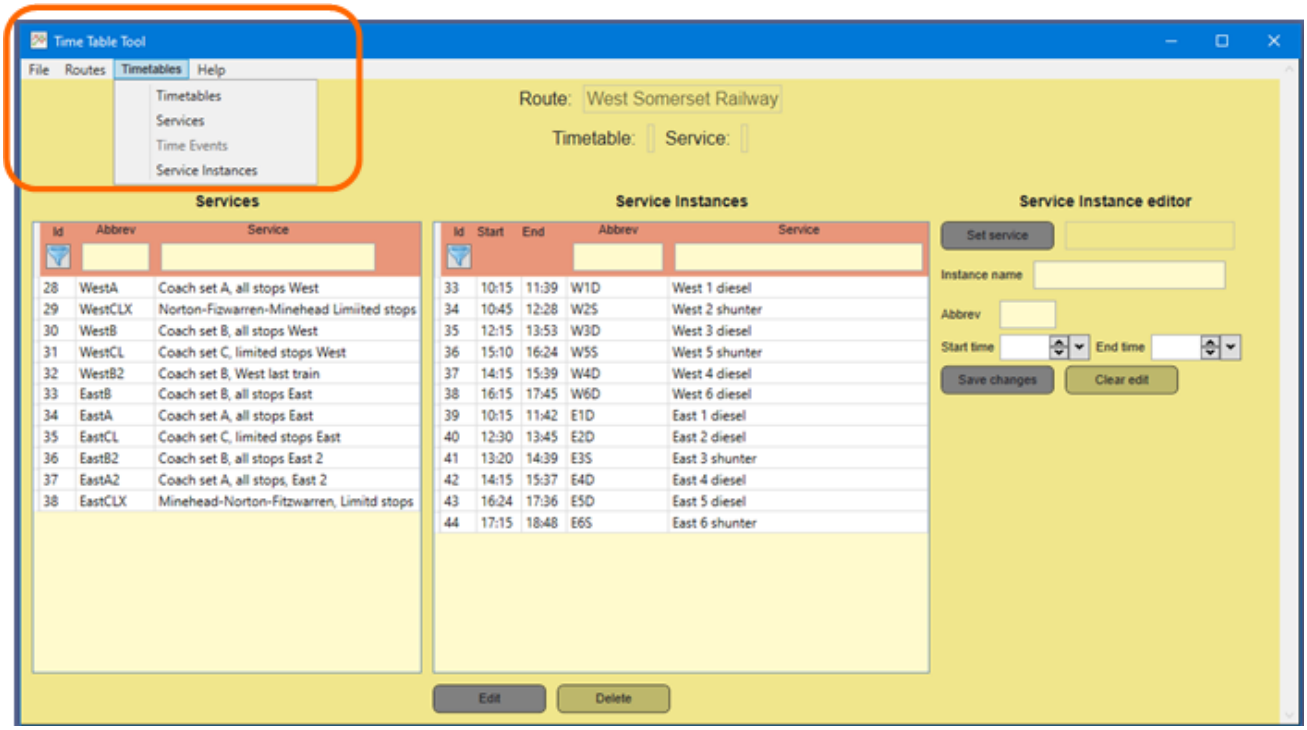


Figure 23 Menu bar

At the top of the main window, you see a simple menu. It has four parts:

File, Routes, Timetables and Help

Each menu has one or more menu-items. If they have light grey text, the item is disabled, because not all conditions are met to use this item. Further on in this reference manual, these conditions will be specified.

6.2.2 Information panel

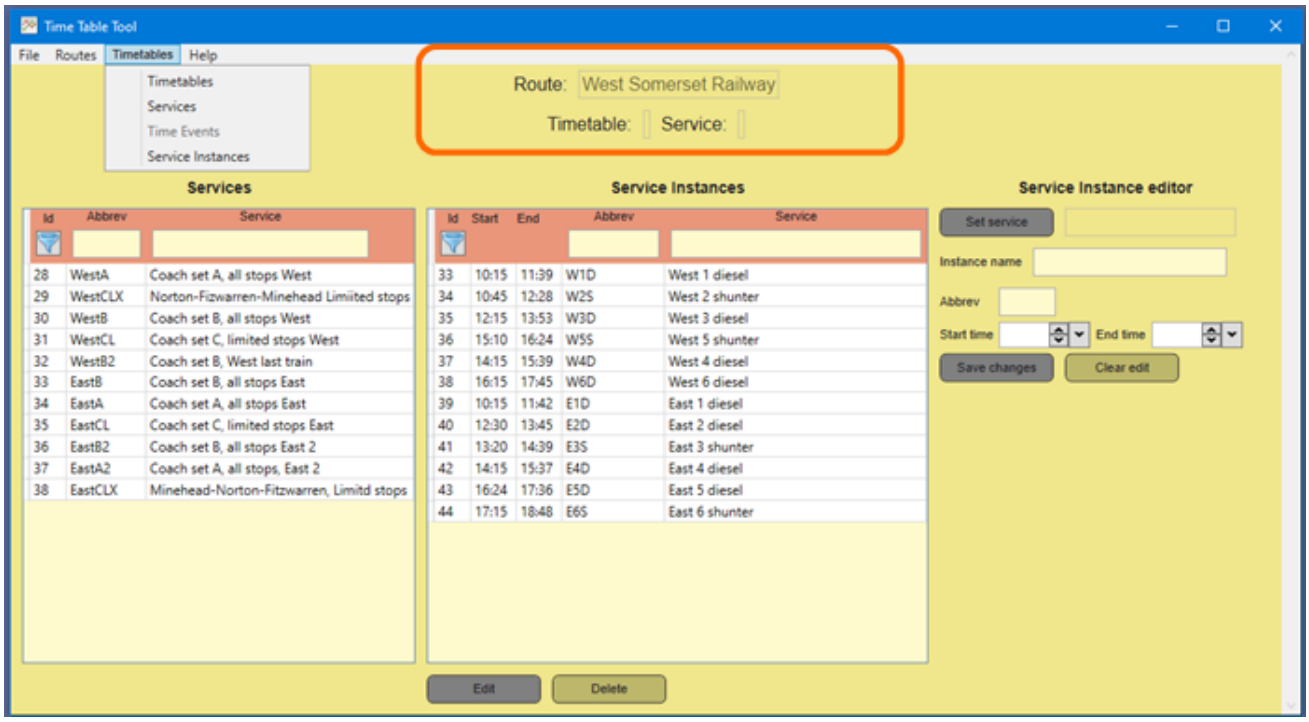


Figure 24 Information panel

The information panels helps you to keep in view what you selected. It is not so beautiful, but just functional. It shows three things:

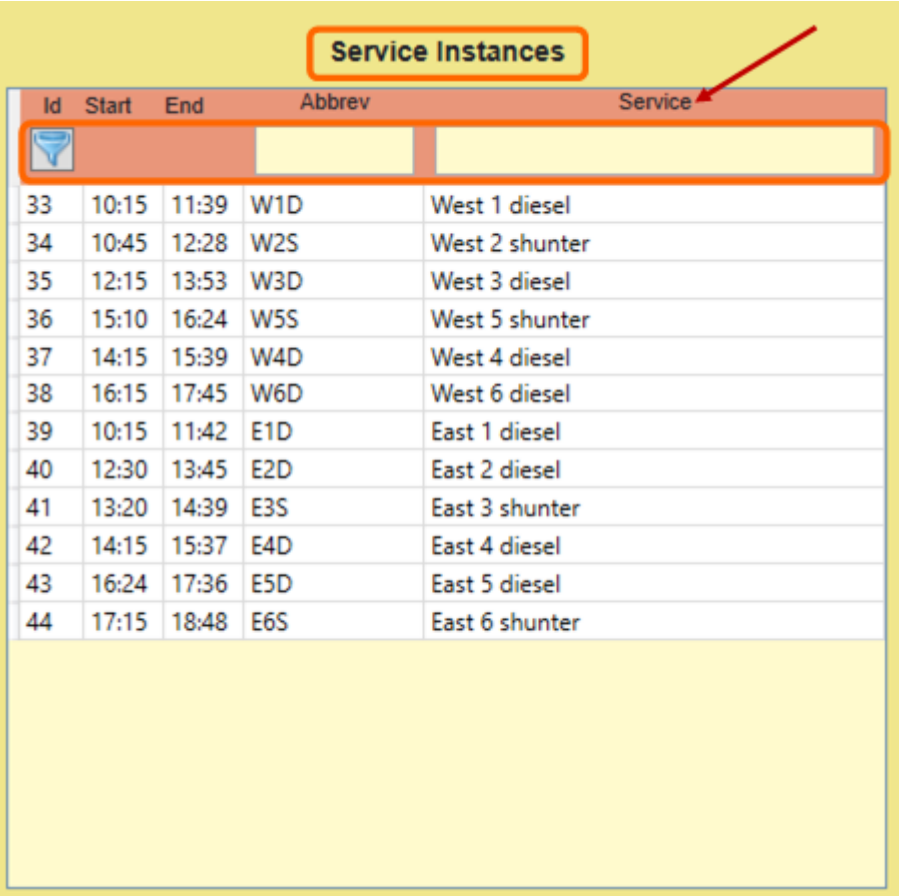
- The selected route
- The selected service
- The selected timetable

In the example, only a route is selected. Both other items do not show a name for an item.

This is useful, because it helps you to understand why an menu-item is disabled and it may prevent some surprises.

I am not yet happy with this solution, so I am open for a better idea.

6.2.3 Tables



Id	Start	End	Abbrev	Service
33	10:15	11:39	W1D	West 1 diesel
34	10:45	12:28	W2S	West 2 shunter
35	12:15	13:53	W3D	West 3 diesel
36	15:10	16:24	W5S	West 5 shunter
37	14:15	15:39	W4D	West 4 diesel
38	16:15	17:45	W6D	West 6 diesel
39	10:15	11:42	E1D	East 1 diesel
40	12:30	13:45	E2D	East 2 diesel
41	13:20	14:39	E3S	East 3 shunter
42	14:15	15:37	E4D	East 4 diesel
43	16:24	17:36	E5D	East 5 diesel
44	17:15	18:48	E6S	East 6 shunter

Figure 25 Tables example

There are many tables in this application. They all share common features. Just above the table, there is a title, that says what this table is about.

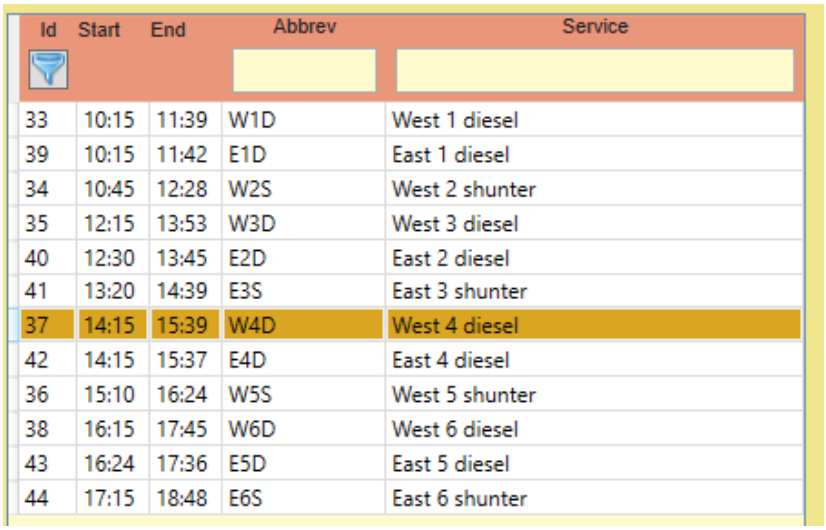
Then, the second marked area, is intended to filter the contents of the table. This not yet doing anything. Worst case it may cause TimetableTool to crash. In future, these filters will be removed or they will be functional.

In between is the header line. This line shows what you find in which column. One of the goodies it has, is that if you click at the header, it will sort the table for you. If you want to sort according to the end times, click at “End”, click again to change the sort order.

The first column shows the record number in the database. Not very useful for you, but it is there.

You can select a row in the table by clicking at it. It will get an orange background then. In the next picture, I sorted the table according to departure time and selected one row.

Generally, you may try to edit directly in the tables. This does not have any effect. I should have disabled this consistently, but that is not always done so.



Id	Start	End	Abbrev	Service
33	10:15	11:39	W1D	West 1 diesel
39	10:15	11:42	E1D	East 1 diesel
34	10:45	12:28	W2S	West 2 shunter
35	12:15	13:53	W3D	West 3 diesel
40	12:30	13:45	E2D	East 2 diesel
41	13:20	14:39	E3S	East 3 shunter
37	14:15	15:39	W4D	West 4 diesel
42	14:15	15:37	E4D	East 4 diesel
36	15:10	16:24	W5S	West 5 shunter
38	16:15	17:45	W6D	West 6 diesel
43	16:24	17:36	E5D	East 5 diesel
44	17:15	18:48	E6S	East 6 shunter

Figure 26 Table with selected row

6.2.4 Buttons

Figure 27 Buttons below a table

At each page, under one of the tables you will find two buttons. The left one is the edit button. It will be enabled once you selected a row in the table. It picks up all data from the selected item and hands it over to the edit part of the window.

The delete button will delete the selected item, including anything that relates to it.

6.2.5 Editor

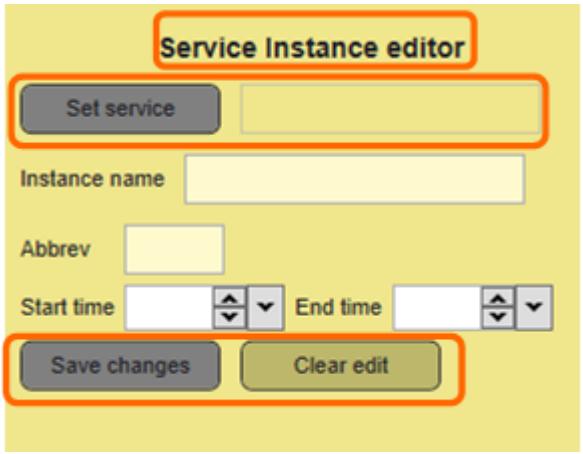


Figure 28 Editor part

In most cases you will find the editor part at the right side of the screen. What it looks like, depends a lot upon what you need to do. The top part is constant, it says what you can edit here. The bottom part is also standard. The Save changes button will be enabled once you completed all required fields. It does what it says, save all changes to the database. It also will clear all fields in the editor. The Clear fields button also clears all fields, but does not save anything.

At the second line of the edit tool, you may see a button labelled Set …. (Set service in the example). This button will be activated if you select a row in the left most column. Then you should press the button and in the field next to the button, your selection will appear. In this way you connect the item you are editing to another item (in database terms for the tech experts: you set a foreign key).

You will in most case need to complete both the name and abbreviation field. For some tables, also a description is required.

At this particular screen, you can set two time values, start time and end time. This control allows just to type it in the format hh:mm You can set the cursor at the hours and the change the hours. At the right, you can select a pre-set time with accuracy of 30 minutes. I do not like this control very much and it definitely will be replaced later this year.

6.2.6 Services editor

In TSW many services are repeated a number of times. For RSN, only six service templates are used. For the North East Corridor, same issue. Therefore a shortcut is created that allows you to add a large number of similar services with a few mouse clicks.

But, be careful. If you blow it (as I experienced), you have a lot to clean up again. In Figure 29 is shown how it works.

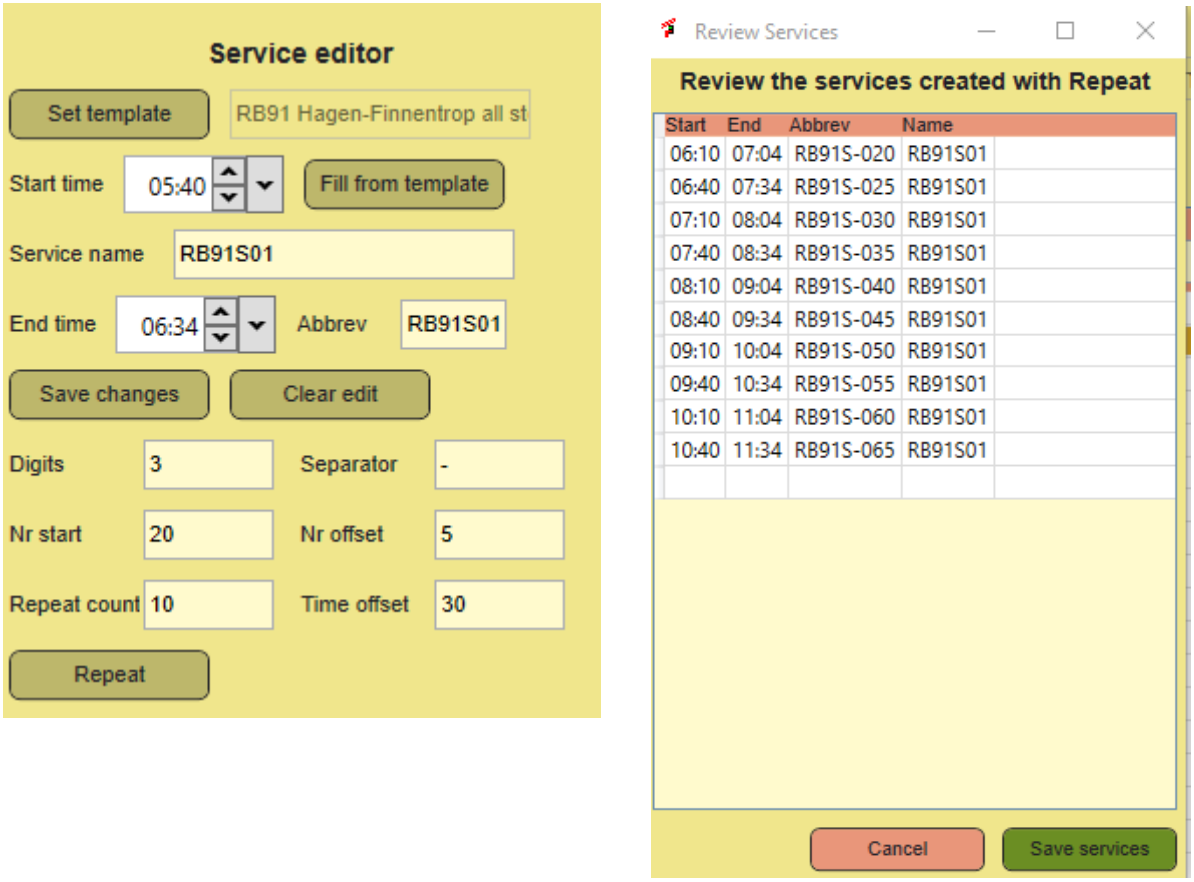


Figure 29 repeat function for services

First, you need to create a service. Then select the service and press the edit button.

You can now work with the bottom part of the edit area, below the Save changes button.

The first line gives you some control on the numbering of the services. It uses a structure like this:

```
{ServiceTemplateAbbreviation}{Separator}{nnn}
```

- In this case, the Service Template Abbreviation is RB91S. This comes from the Service Template.
- The separator, you can set. As a default a dash is set.
- Finally, a sequence of digits is added. At the digits field, you can set the number of digits, where 3 is the default value.

The second line has two parameters, a start value. That is the number the first service you create gets. The second number is the offset for each next number. In this example, the first service will get number 020, the second 025 etcetera. This allows you to mix different service types that form some kind of sequence together.

The third line completes the specification: The repeat count determines how many services must be created. Finally the Time offset, determines how many minutes after the currently selected service the next one must be scheduled. In this case, The selected service starts at 04:40, so the first repeated service will depart at 06:10 (30 minutes later).

The you press the Repeat button. To prevent you from doing stupid things, a popu will be shown, tah shows all new services. If you press Cancel, noting happens. If you press OK, 10 new services will be created. For

NEC I created up to 20 services at once, which makes a significant difference for the amount of work define a time table.

6.3 User settings

In the File menu you can find a menu-item **Settings**. At the moment you can define the size of the graph here. It is recommended to make it as big as your screen allows.

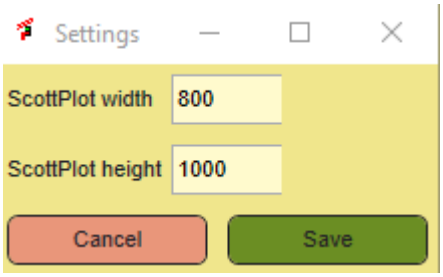


Figure 30 User settings dialog

This uses a dialog screen. You must close it before you can do anything else. The data is stored in the Windows registry (key: **Computer\HKEY_CURRENT_USER\Software\Holland Hiking\TimetableTools**)

Note: there is no protection against doing anything stupid here. If you set the values too low, you may no longer see the graph. I will fix this someday. Till that moment, be careful.

6.4 About window

The help menu has three functions at the moment. The first to mention is the About window. This window pops up if you select the menu-item and shows a nice screen with some useful information.

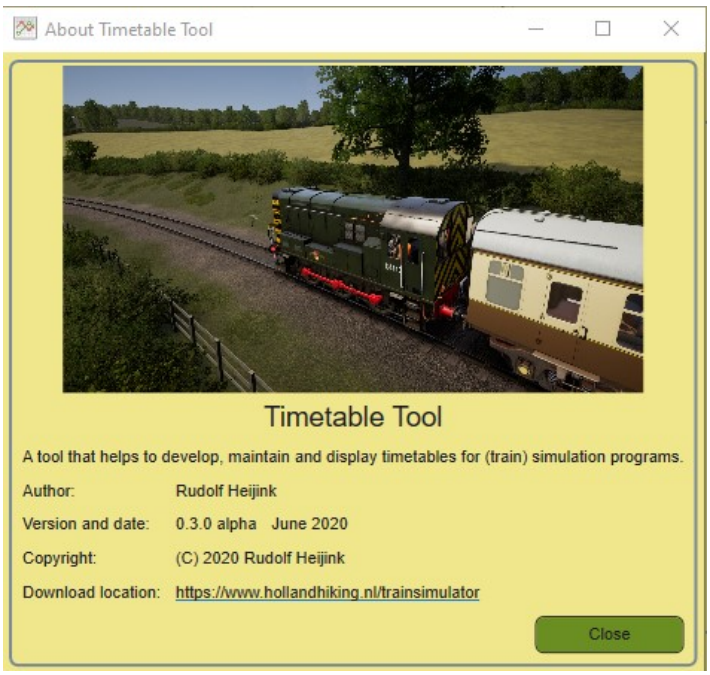


Figure 31 About window

It shows the actual version number and the URL of the website where you can download updates. Here you will find a donation button soon …

You must close this window before you can continue.

6.4.1 Manual

The second menu-item in the Help menu is the Manual menu-item. Guess, this will display this manual. It will only work if you have setup a pdf viewer. Normally this should not be a problem. TimetableTool is not aware of the pdf reader you prefer. It will just use anything you prefer.

6.4.2 Logging

The last item in the help menu is the Logging. If you click, a window is opened that shows all log messages. A log message is created in most cases if an error occurs, or just to inform you about something that happened. In most case, you also will see a blue popup screen at the top of your window.

Figure 32 Popup window

After a few seconds it will disappear again, but the text can be reviewed in the Logging view. The logging view is modeless, which means you just van leave it open and continue to work.

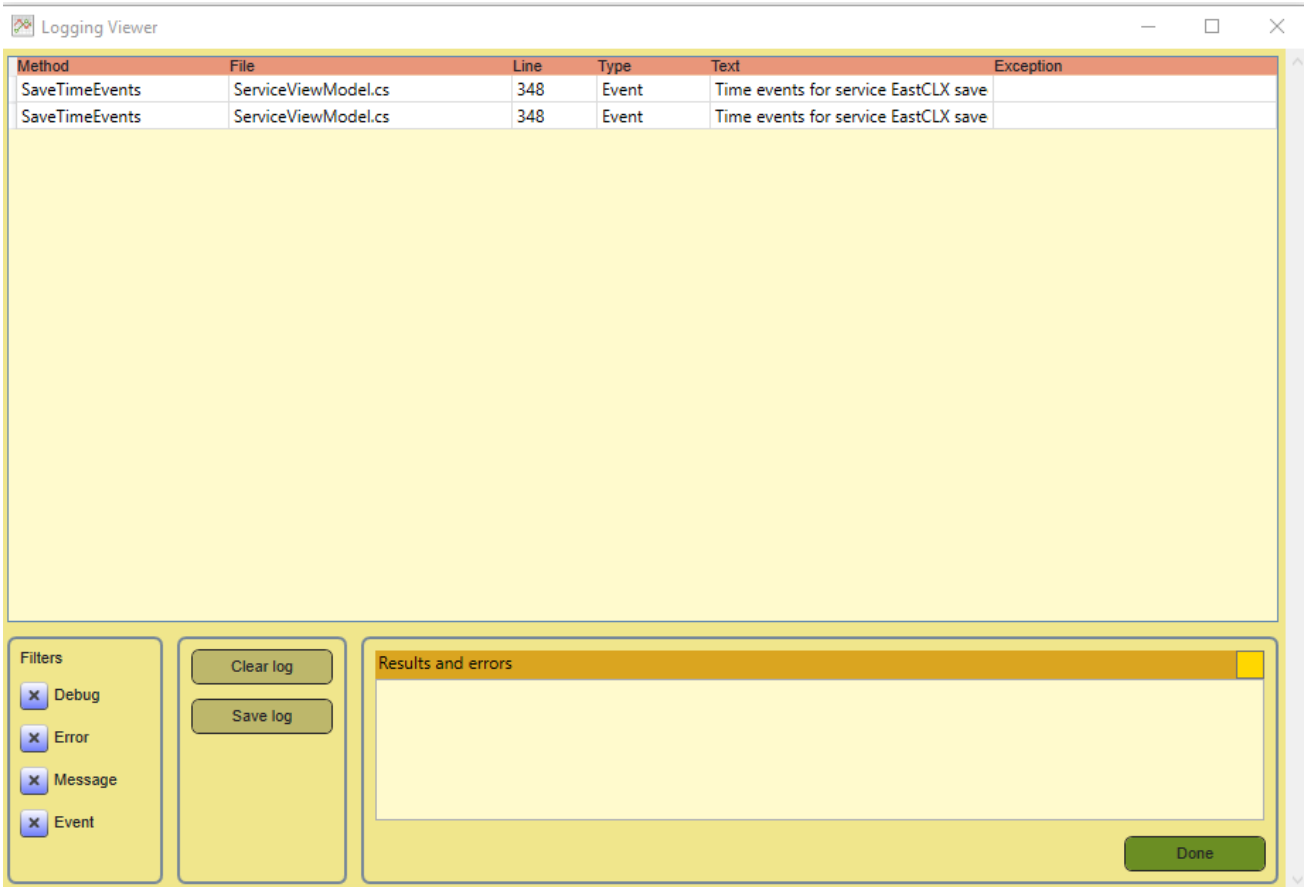


Figure 33 Log viewer

At the top you see the familiar table, showing all entries. At the bottom left, you can turn logging on or off in this table. The log will be there, but is not shown.

The Clear log button clears everything in the table. The Save log button will export the log data as csv file.

If you experience any problems, please make an export and send it to me. This is very helpful in finding issues.

6.5 Viewing ScottPlot graphs

As stated before, there are three ways to look at a timetable. The first one is the classic timetable booklet view. In order to make this work, you need to have selected a route and a timetable, where the field service direction is set. The second one displays arrival and departure times per location.

Finally, you can show all services in a time graph as it is used by professional railway timetable designers. In the current version, there is the brand new ScottPlot graph. The latter one will be further developed. I intend to abandon the original one.

There is not much to add to this information for the classic timetable and the arrivals and departure timetables. For the ScottPlot timetable, there is a lot more to say. This graph is named after Scott harden, who developed the free graphics package that makes it possible to create these fantastic graphs.

The first thing to cover is on how you can change your view to find the information you need. For the West Somerset Railway, there is no issue. Only 25 services, which you can distinguish easily. In this section we will use the TSW Ruhr-Sieg Nord timetable, which has about 130 services. This still is a modest number.

When open this graph it looks like Figure 35. This is completely useless, so we need to do something.

The first thing I suggest is that open the Settings (see 6.3) and try to enlarge the plot area, so that the window just fits on your screen. I built the window in such a way that will not take more than 90% of the screen, to make sure you sill can reach the sizing control. If you make the plot larger than this, it will not fit and you may see scrollbars on the window. It works if you do so, but it is better to avoid this, because you will lose sight on the horizontal axis.

To give you some guidance: 1200x1200 is a good setting for a 2560x1440 screen. You can still increase the width if there are many stations. In the next version I will make the X-axis more compact.

By default, the graph represents 24 hours. If your services do not run during the night, you time axis will represent less hours. You can restrict this manual, by setting the interval at the top left. It may be obsolete, it was my first attempt to make it more manageable but it is still there. What it does: It will filter you timetable and only include services withing the set interval.

Now we come at the step that does the real trick. At the top left you see two sliders, zoom and scroll.

The zoom slider will enlarge or reduce the time scale. You will never see more than 24 hours, you cannot zoom out. I set the maximum value for zooming at 10. Let me know if this is OK for you. If you zoom in, part of the graph will get outside our view.

To fix this, use the scroll slider, to bring the time frame you need to see back into view.

In you what happens. I set the zoom to 10 and the scroll to -4. This shows the services between 06:00 and 07:00. Now you can clearly observe the interactions between all services in this timeframe.

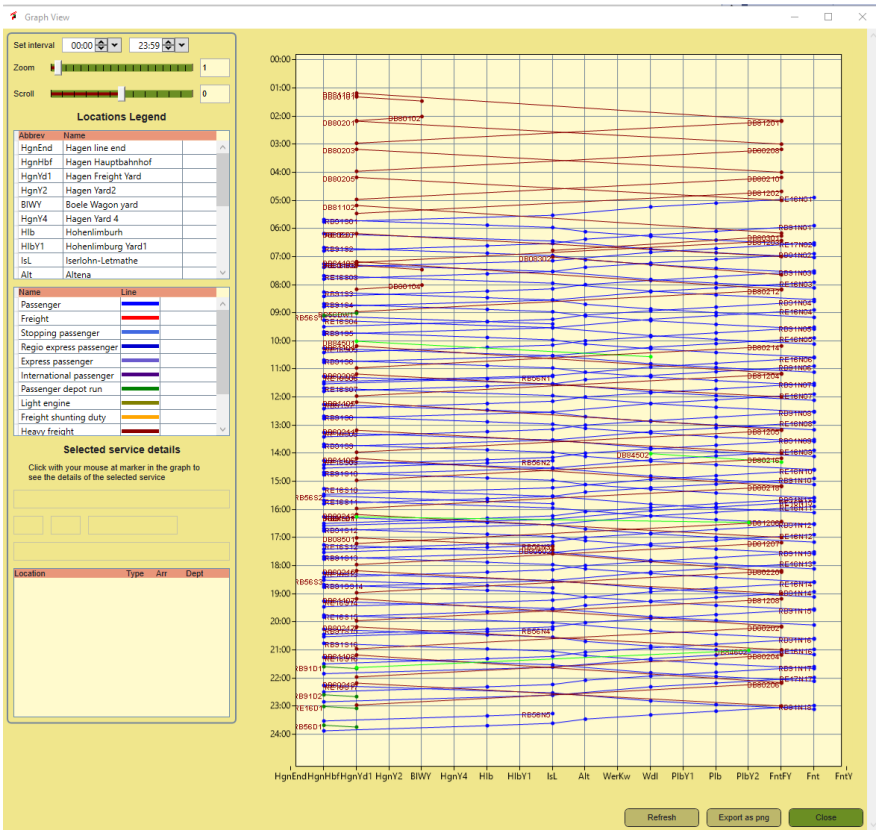


Figure 35 Timetable graph for Ruhr-Sieg Nord

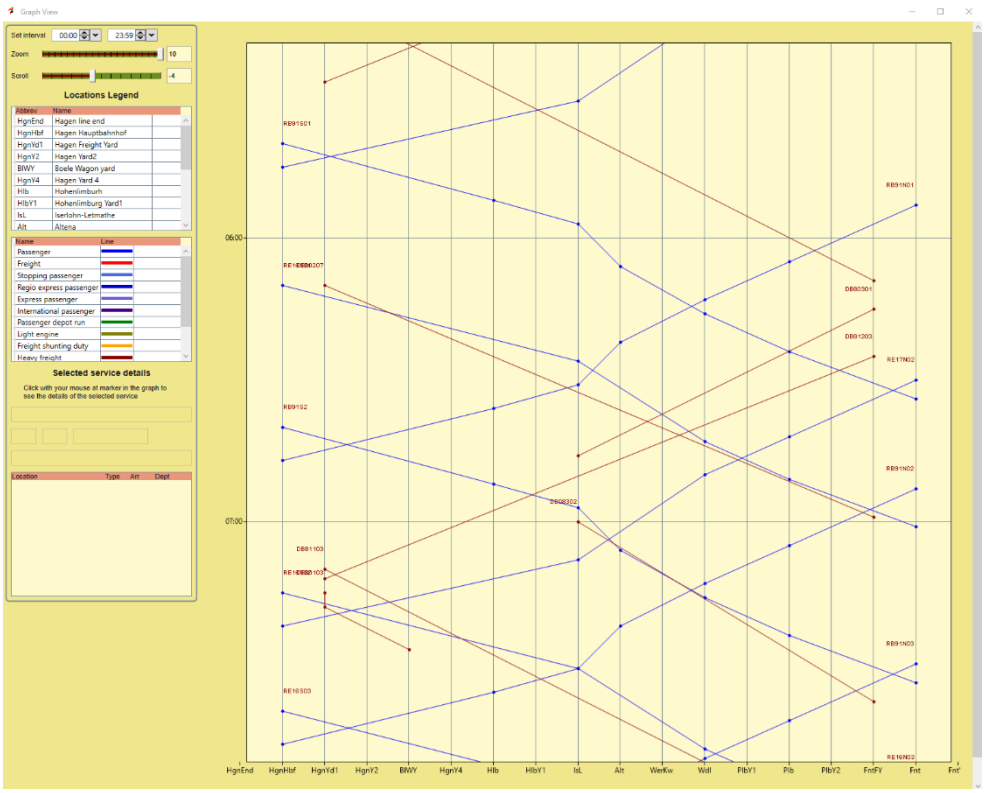


Figure 34 Zoomed graph view

You may see the different line colours. These represent the service classes. At the left you see a legend that explains the colour scheme. Here you also find a legend for the location names.

In the graph you see dots representing the locations. Now put your mouse cursor near one of these dots and left click. You see now several things happen. The line width will grow, to highlight the service you selected. You also may notice a red circle around the point you selected. This does not have any significance, but it is helpful for the development.

Also, you see in the left panel the details of this service, including service class and timetable. It is is not yet possible to edit the service or the service template from here. I have not yet decided if this is a handy option.

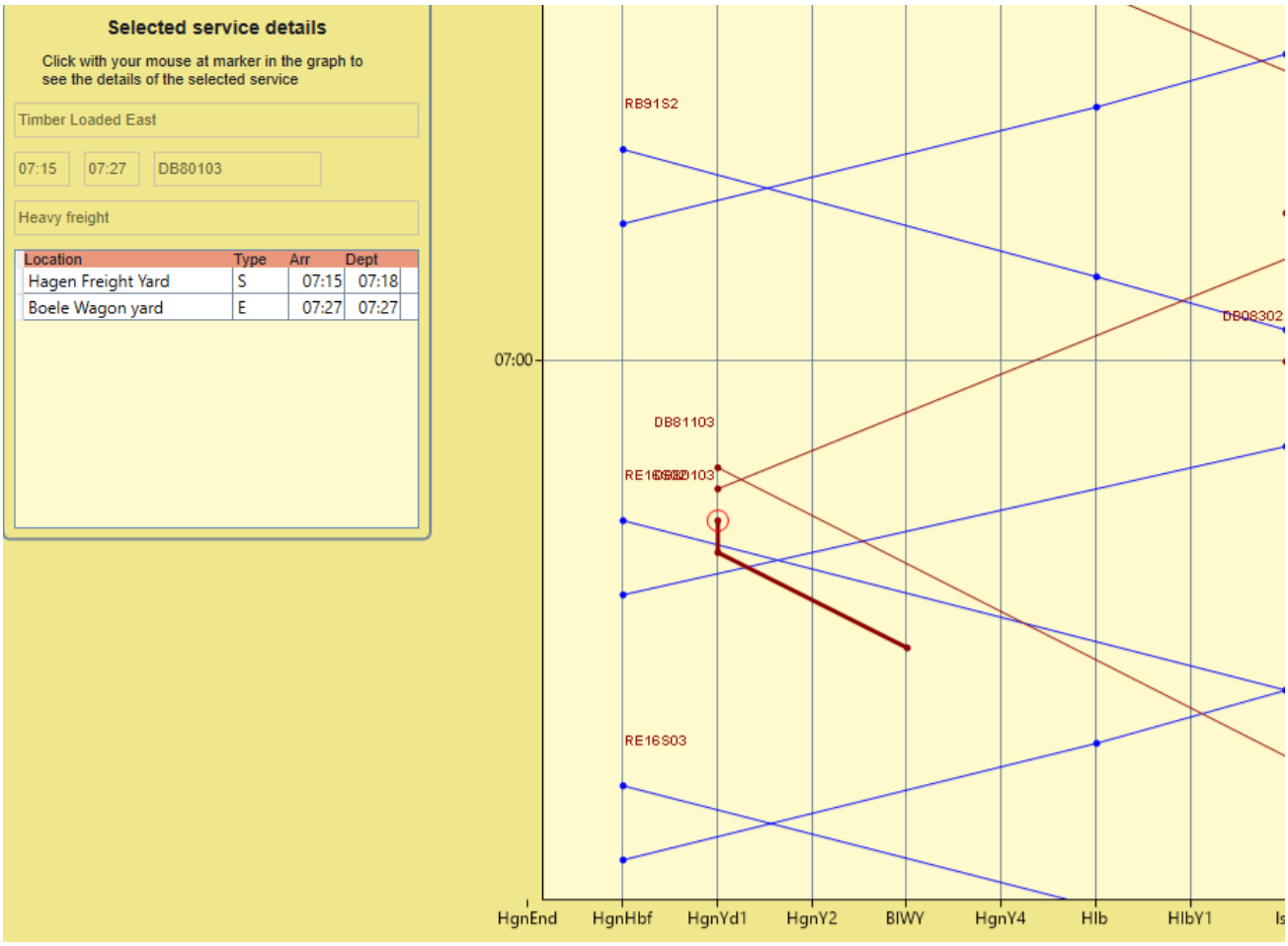


Figure 36 Timetable graph with a selected service

6.6 The database

All TimetableTool data is stored is a relational SQL database. The good news is that this database is a stand alone tool and does not require any complex setup. There is a tool that gives you direct access to the database. You can download it here:

https://sqlitebrowser.org/

You will probably recognize the database structure in the sections of the tutorial.

Below, you see a simplified overview showing how data is organized:

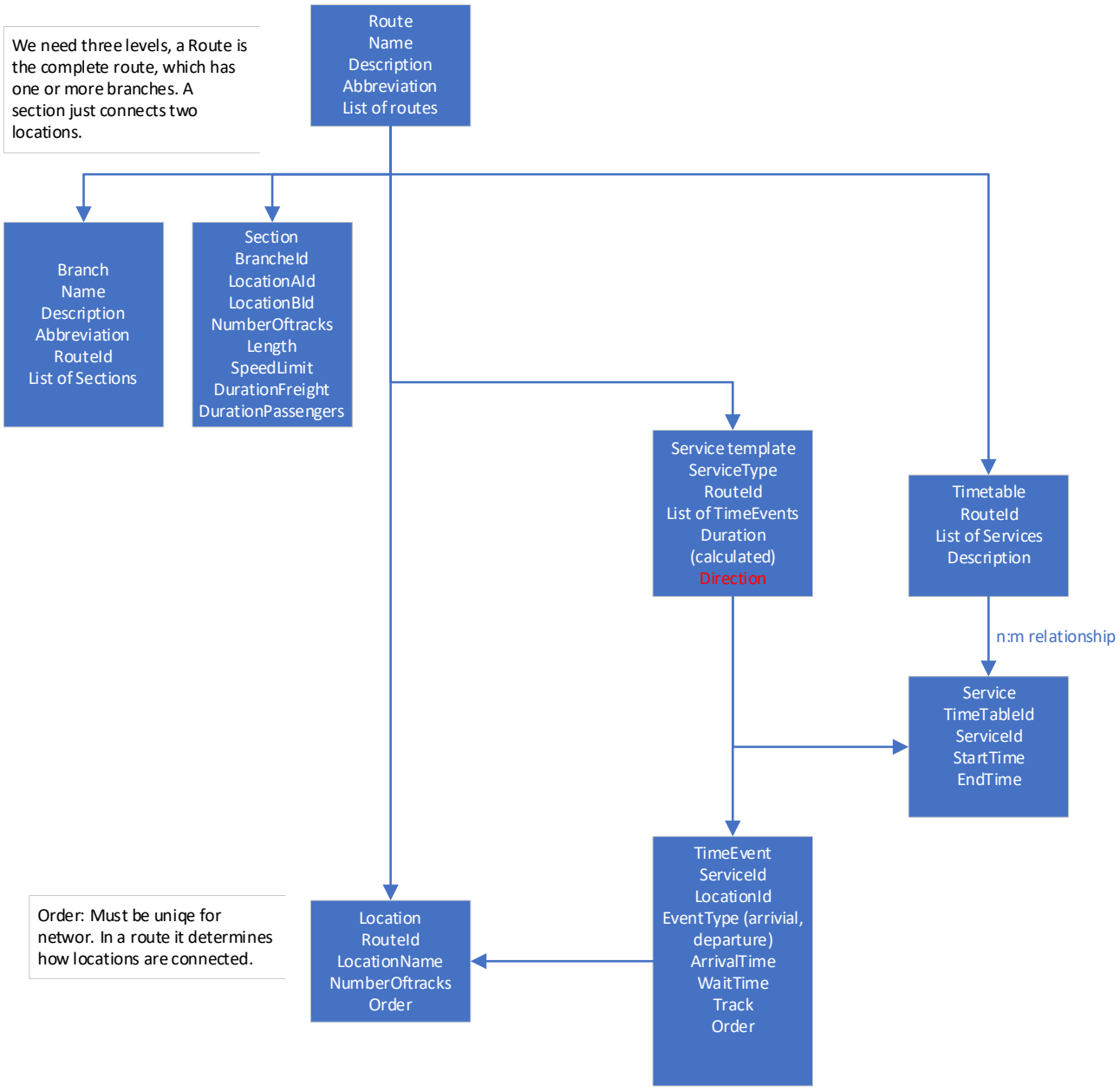
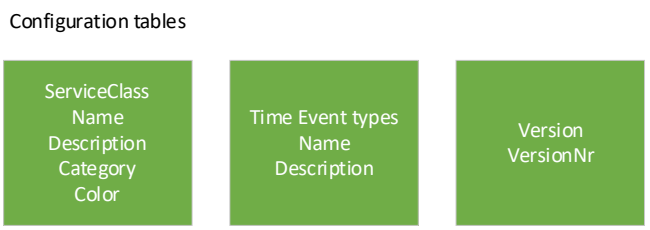


Figure 37 Database structure

There is one tricky point. Because services can be connected to multiple timetables, a connector table is inserted here, to establish the n:m relationship. (If you do not understand this, don't worry but in that case this chapter is not for you yet).

Besides this data structure, there are some configuration tables:



The datamodel does not use foreign keys to look up configuration stuff, but just does a text search.

Please do not try to modify the configuration tables. Your changes may be overwritten any time. If you feel the need for additional values, let me know and I will see if this possible.

You can add views to the database or experiment with extensions to TimetableTool. I am looking forward to your proposals!

6.7 Backup and restore

The backup and restore function does NOT protect you against crashes of your disk. You should include the data folder for TimetableTool into your backup procedures.

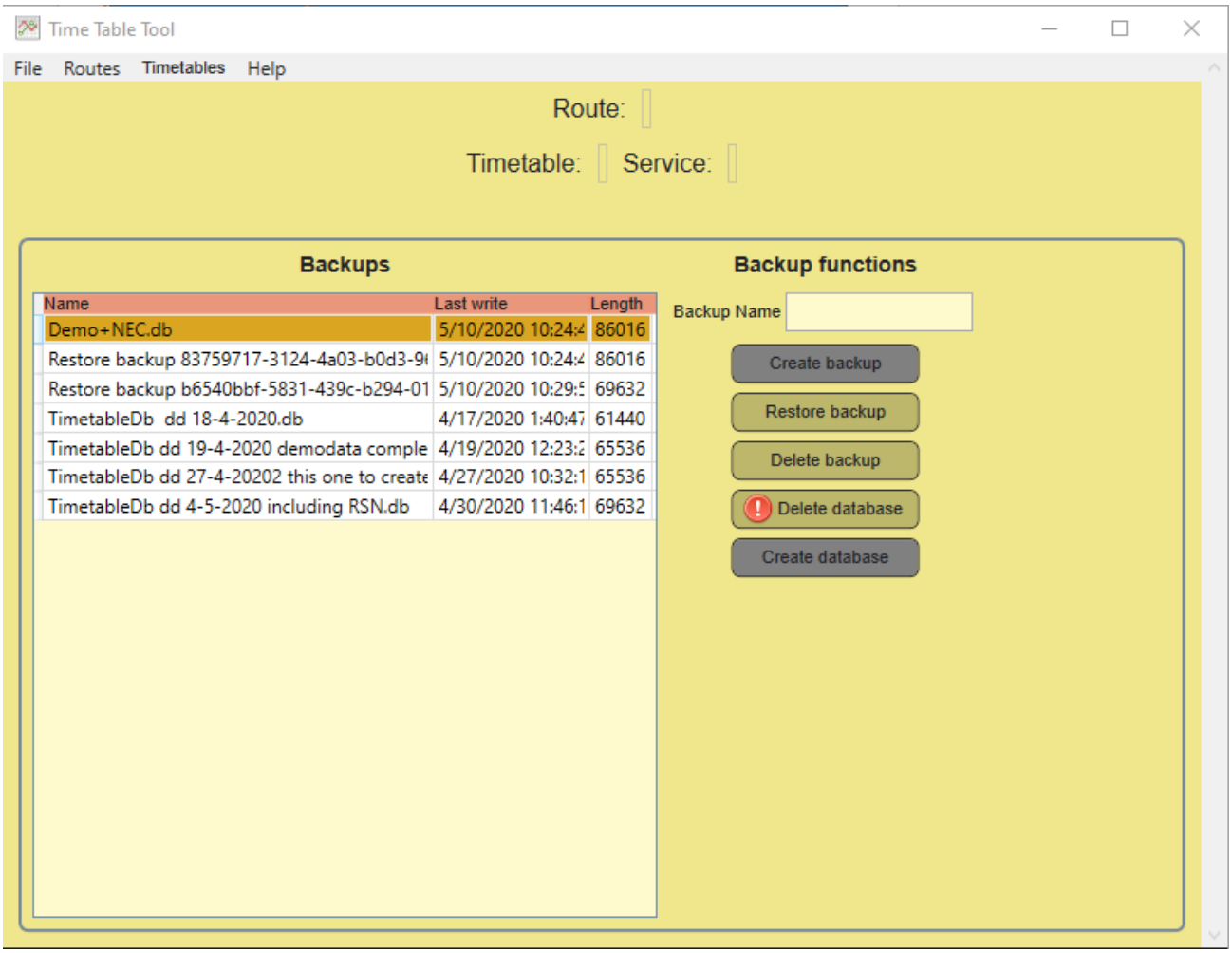


Figure 38 Backup window

In the File menu, click at the item Backup and restore to open the backup function. You can do this any time, because the database connections are closed after each request.

Backups are stored in the folder Backups in the TimetableTool data folder (normally in the Documents folder).

The available backups are shown in the table at the left side of the window. At the right side, you see five backup functions:

- **Create Backup**. This makes a new backup. It is enabled once you have defined a filename for the backup in the Backup Name filed.
- **Restore backup.** This retrieves a backup and makes it the active database. Your existing database is automatically saved. I do not want you to loose valuable information by a split second mistake. The name for this backup is generated automatically.
- **Delete backup.** This function deletes an existing backup. It does not asp for confirmation an there is NO restore.
- **Delete database.** This will just delete the active database. It will make a backup automatically, but does not issue any warnings. So this is a bit of a dangerous operation.
- **Create Database.** This function will create a new active database. You need to delete the existing database first. The new database will contain the two test routes.

If you restart TimetableTool after deleting a database, automatically a new database with test data will be created.

It is recommended to make backups at regular intervals. In the table you may notice that the last time the database was written to disk is shown. This may help you to find which version you need to restore.

An alternative is export your routes at regular intervals. This export will contain all data and you can even share the export with other people. It gives a more fine grained backup facility.

6.8 Export and import of routes

There is a export function, that will export data from the database in a .csv like format, though I use the .ttt file format. You can import these data in another database, where the relations between the tables will be set properly.

You should NOT try to edit .ttt files manually. This may corrupt your database and it will be a lot of work to get that fixed, if you can …

The export function is located in the Routes window. You must select a route, and then you just press the button.

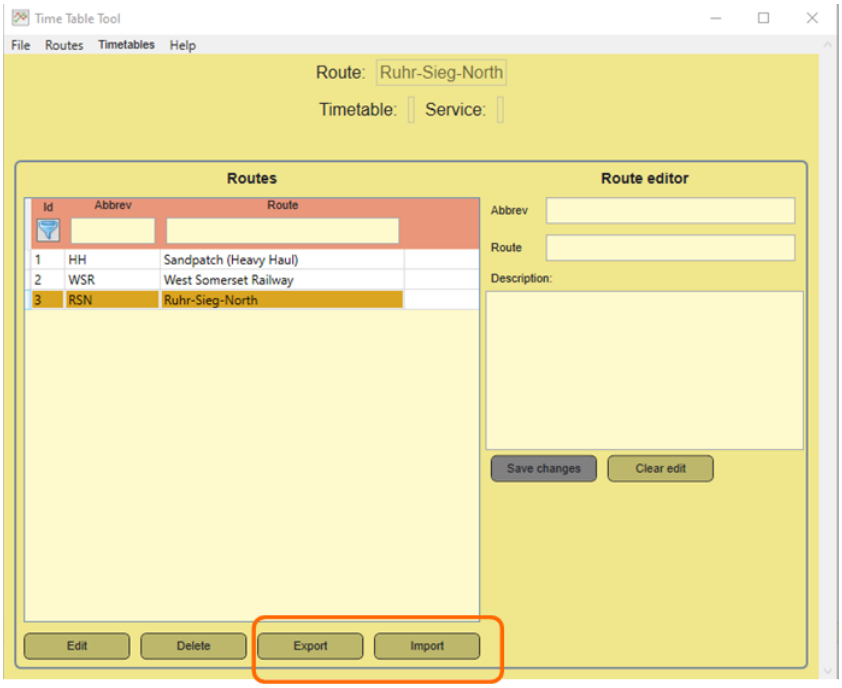


Figure 39 Export and import the TimetableTool database

The export file will be stored in you TimetableTool data folder and is named <RouteAbbreviation>-<Date>.ttt

To import a route, press the import button, select a .ttt file and you are done.

6.9 Source code

The source code for TimetableTool is publicly available at GitHub. If you want to contribute and help to develop code, please contact me by mail.

To do so, you need some base knowledge:

1. C# 8.0
2. MVVM principles (I use Caliburn.Micro)
3. Dependency Injection
4. WPF
5. SQLite
6. Visual Studio 2019

It would be wonderful if you have good knowledge on test automation, using xUnit.

Here you can have a look at the source code and download it:

https://github.com/RudolfJan/TimetableTool

You probably cannot compile it, because a library from another project is used. I need to find out how to make that library available here. We will find a way to solve that.

7 Known issues

Here you find a list with known issues. If you find any other issue, please let me know as detailed as possible. You can send it to this mail address: trainsimulator@hollandhiking.nl

Nr	Issue	Priority	
1	Time picker not working brilliant	Low	Will be solved in 2020, November, when Microsoft publishes .Net Core 5
2	No automatic updates	Medium	Coming soon …
3	Graph does not yet look nice when too many services are shown	Medium	This should be more flexible, probably a scroll viewer need to be added.

A. Links to documentation and tools

All my tools, including TimetableTool, TSWTools and LuaCreator plus my guides are available at the locations mentioned below:

Topic	URL
Holland Hiking	http://www.hollandhiking.nl/trainsimulator/index.php
For donations	https://www.paypal.com/cgi-bin/webscr?cmd=_donations&business=LNBS2R49HHBF6&currency_code=EUR&source=url