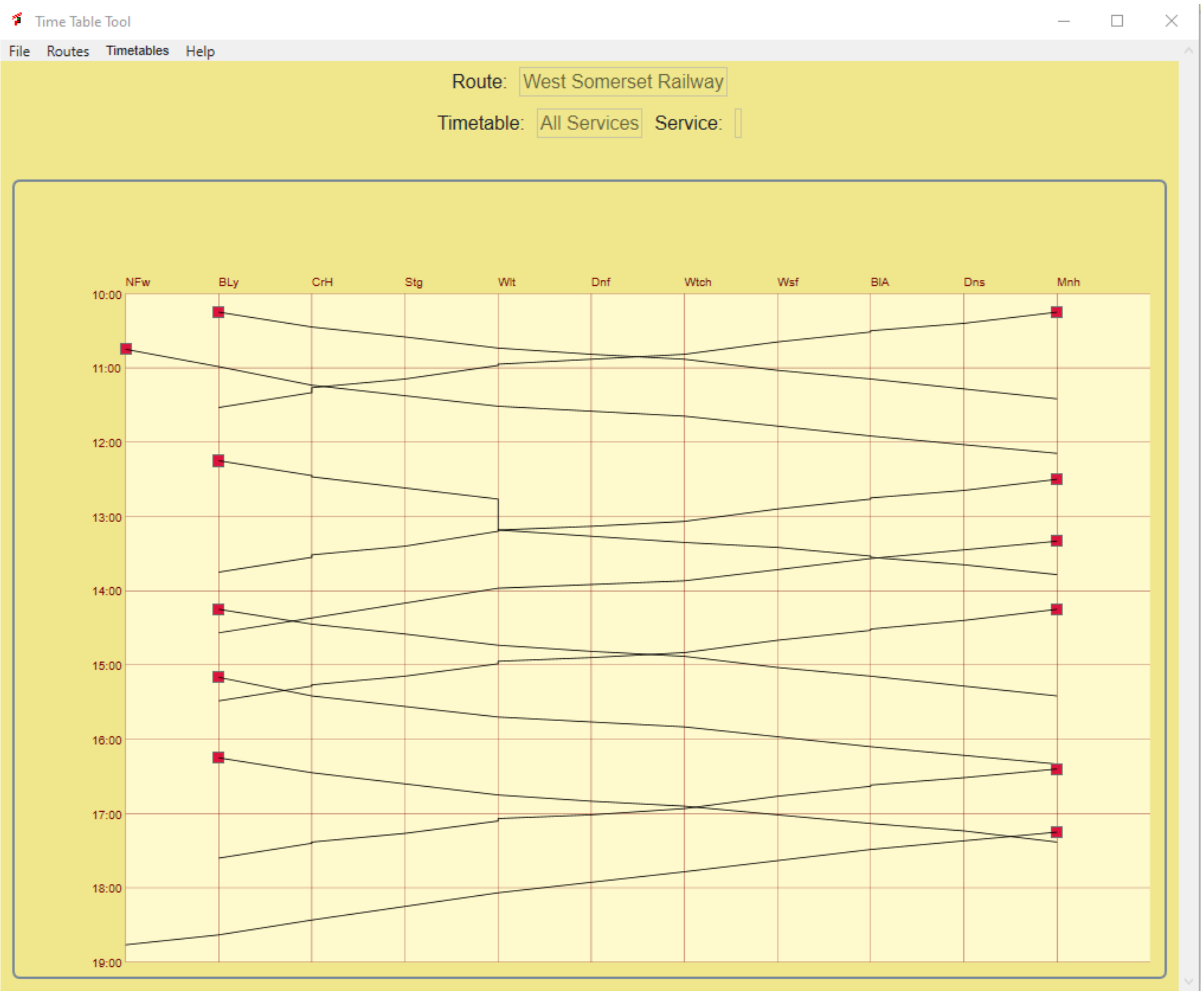


Route: West Somerset Railway

Timetable: All Services Service:

TimetableTool Manual

Rudolf Heijink

Version 0.1, May 2020

Copyright © 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.
- 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

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

Frontpage image:

<Description>

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

List of figures

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.

This first version is still a bit primitive. It works, but it is not yet optimized for easy use. I definitely will add a number of shortcuts, but first I need a stable base and demonstrate the principles on how this all works.

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

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

This the initial version. It has all core functions working, but the user interface is not yet great and you definitely will like more views.

2.2 Next version

I have not yet decided what will be in it. Likely it will be possible to delete timetables. This is not yet supported and I will make life a bit easier by introducing some more shortcuts. I will start playing and design at least one new timetable, to get a better idea on what we may need. Your comments and wishes are welcome, please share!

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. Caliburn.Micro MVVM library
5. Dapper for Database Access
6. Squirrel to create installation packages
7. Word and PDF for the user manual

2.4 Version 0.1

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

3 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:

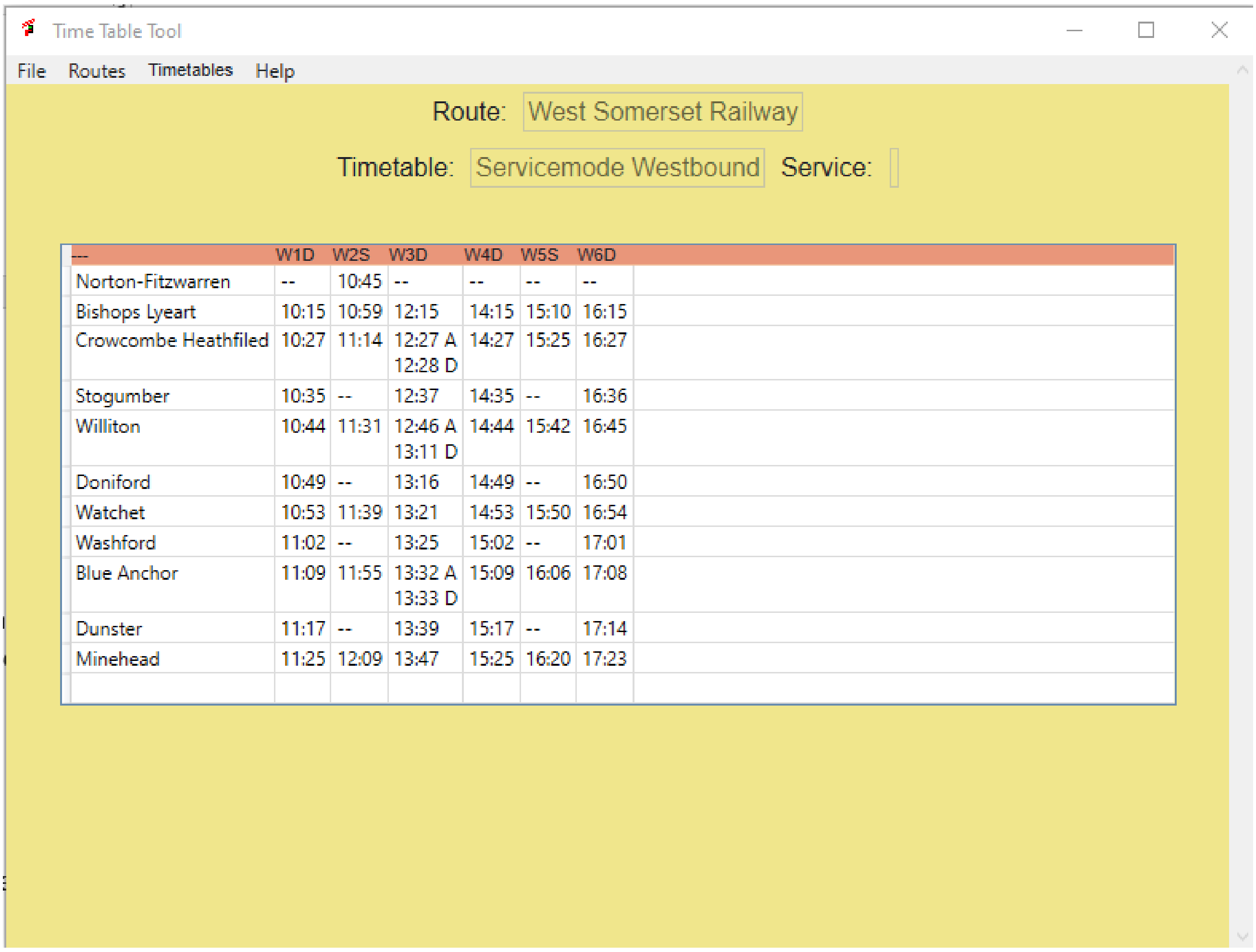
1. The West Somerset Railway
2. Heavy haul (Sandpatch)

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.

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 TimetableToolFigure 1.



Route: West Somerset Railway

Timetable: Servicemode Westbound Service:

---	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 2.

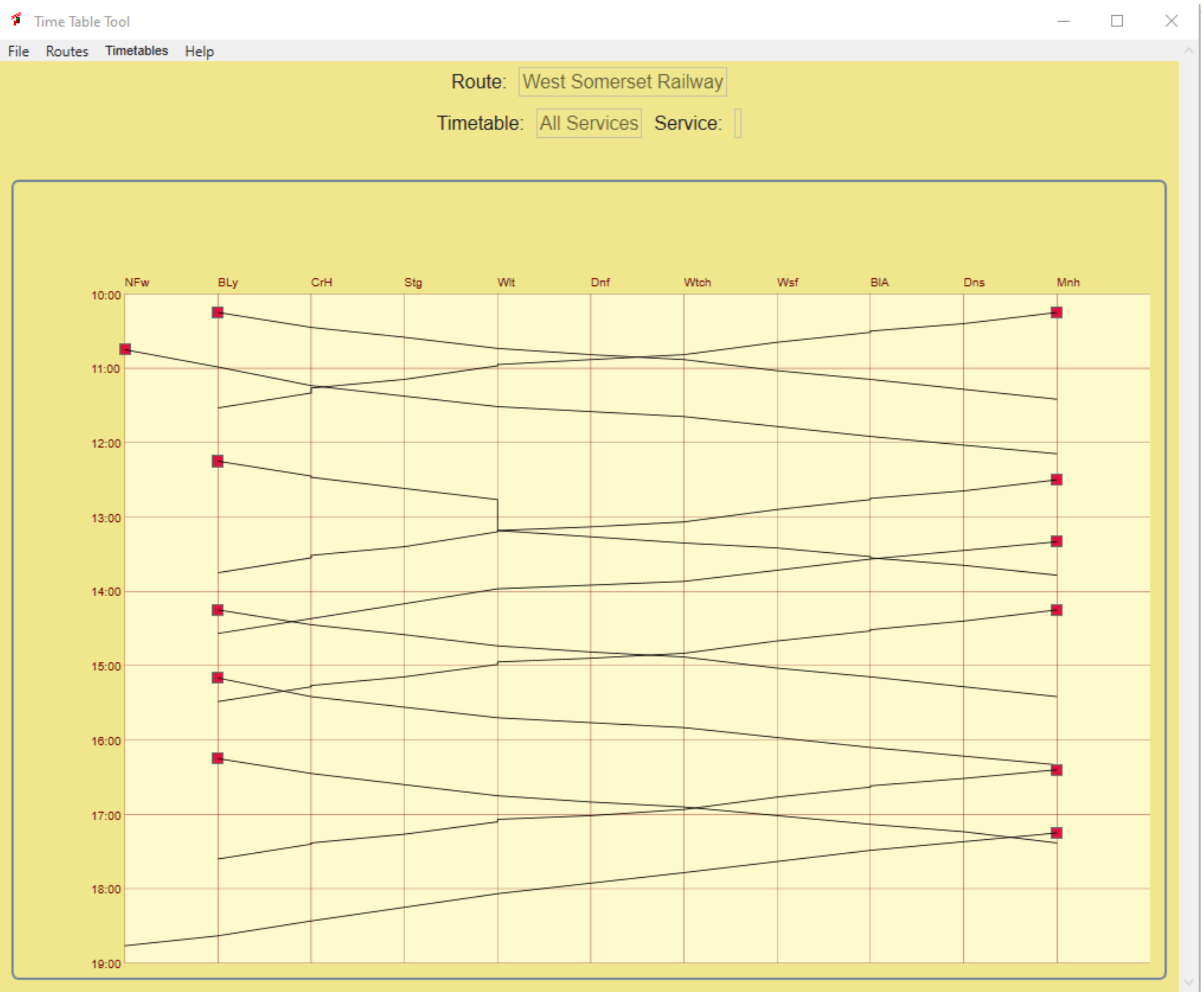


Figure 2 Graph view of the complete services timetable for West Somerset Railway

The red squares denote the starting points. At the top you see abbreviations for stopping locations. If you move the mouse over the red squares (which are buttons), you will see a tooltip with additional information. The same happens if you move over the stopping locations.

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:

- Timetable per stopping location, showing all arrivals or departures
- 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.

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:

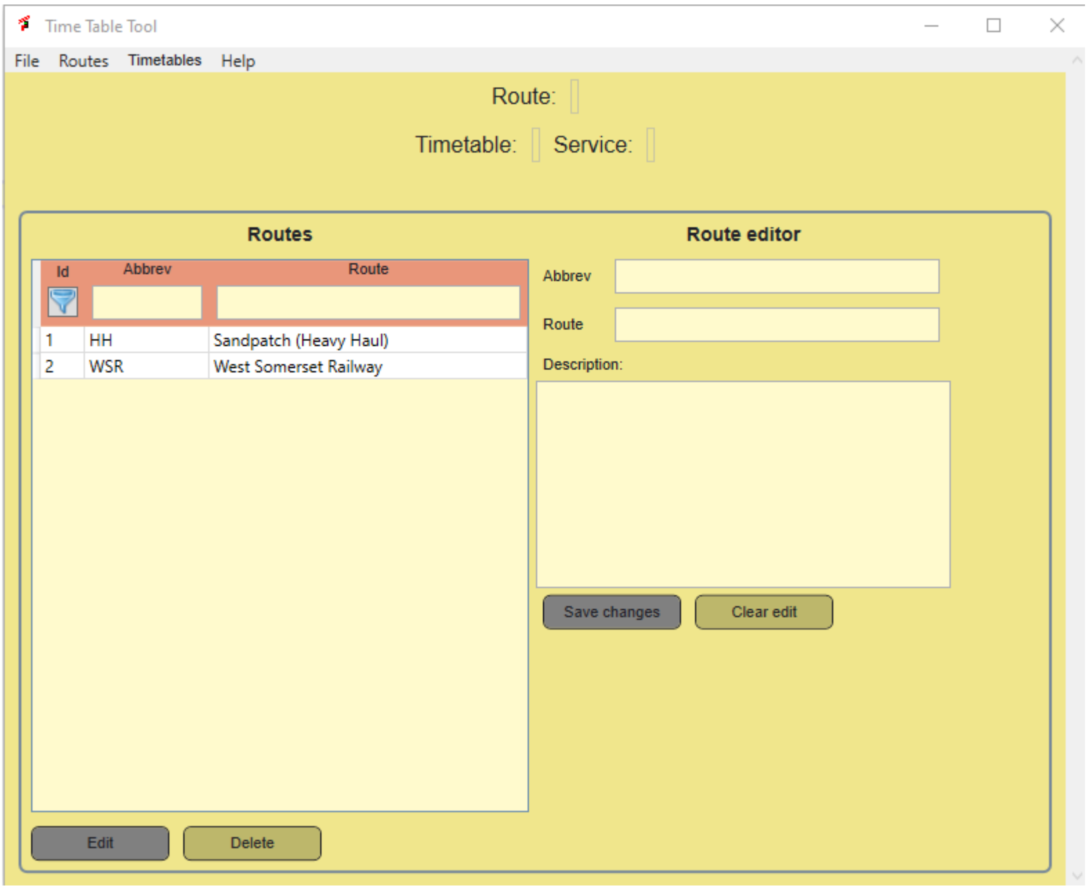
1. Define a **route**. All timetables are bound to a route.
2. Add an number of named **locations** to the route. A location can be a station, a siding or any other point of interest. You also must define an **order** for the locations.
3. Define **directions**. For instance, you may drive from west 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 **services**. A service is a container to hold stops, driving times but not a fixed departure time. This make sit possible to run a service multiple times. Defining services is to most work, so I included some optimizations.
5. A **service instance** makes the service a real drive, by giving a service one or more departure times. In the graphs you see service instances.
6. Bundle a number of service instances 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 3 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.



Id	Abbrev	Route
1	HH	Sandpatch (Heavy Haul)
2	WSR	West Somerset Railway

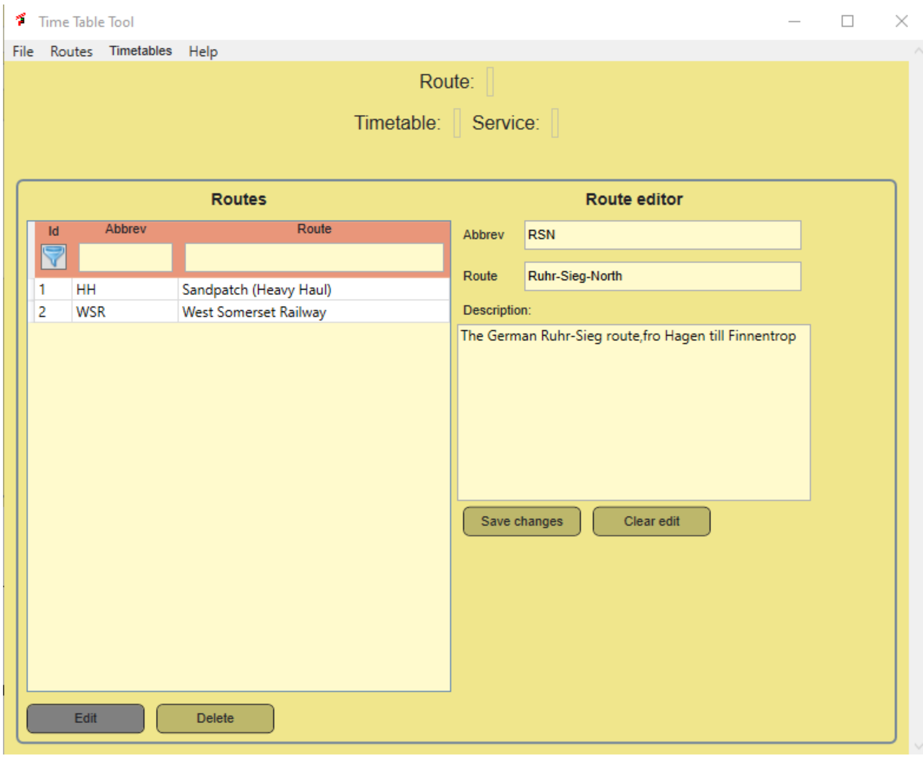
Figure 3 Route selection and entry screen.

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.

The Delete functions have not yet been implemented, so once your route is saved, you can edit it, but you cannot delete it at the moment.

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



Id	Abbrev	Route
1	HH	Sandpatch (Heavy Haul)
2	WSR	West Somerset Railway

Abbrev: RSN

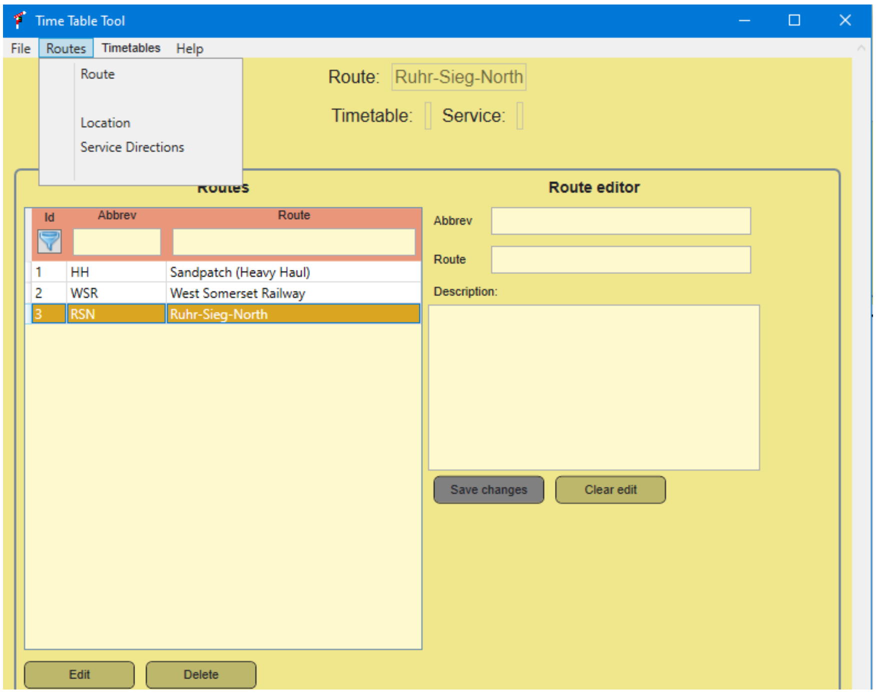
Route: Ruhr-Sieg-North

Description: The German Ruhr-Sieg route,fro Hagen till Finnentrop

Figure 4 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.



Route: Ruhr-Sieg-North

Id	Abbrev	Route
1	HH	Sandpatch (Heavy Haul)
2	WSR	West Somerset Railway
3	RSN	Ruhr-Sieg-North

Figure 5 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.

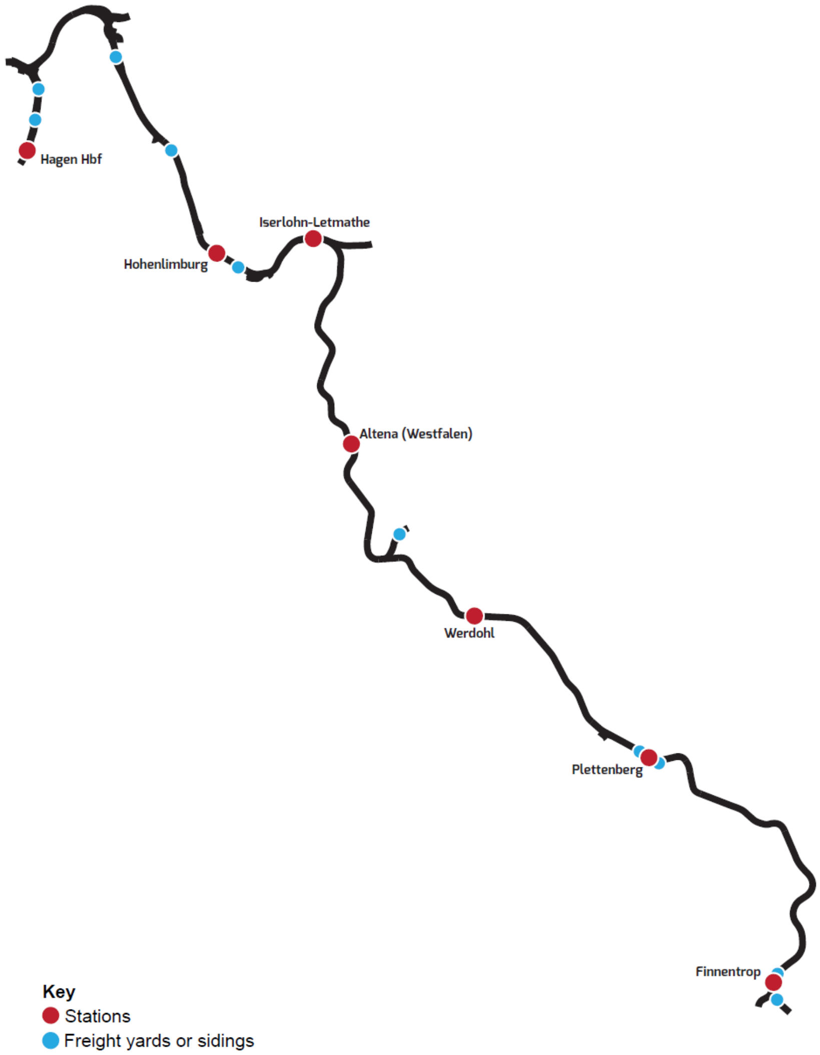
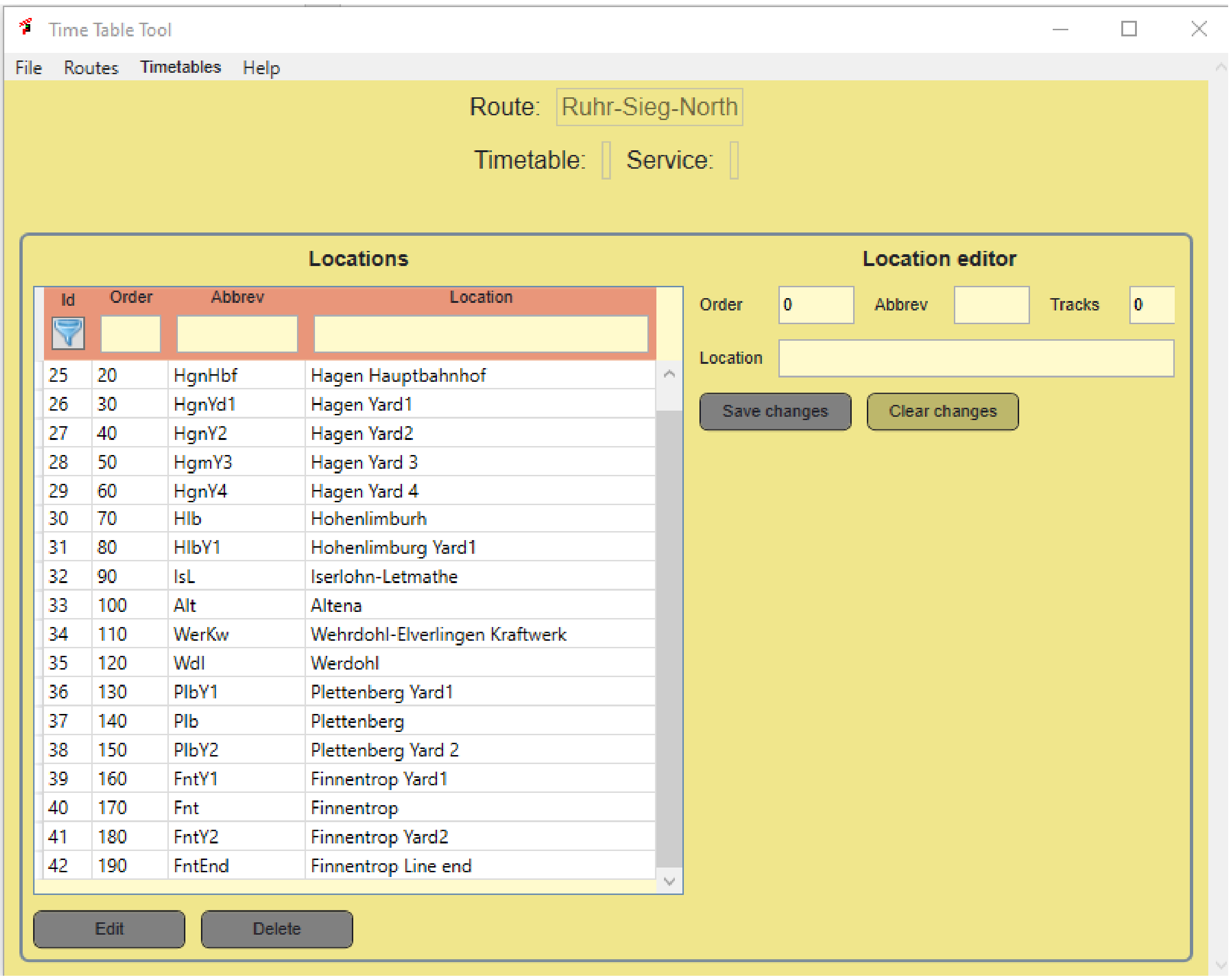


Figure 6 Route layout as shown in the official game manual

In Figure 6 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.



Route: Ruhr-Sieg-North

Timetable: Service:

Id	Order	Abbrev	Location
25	20	HgnHbf	Hagen Hauptbahnhof
26	30	HgnYd1	Hagen Yard1
27	40	HgnY2	Hagen Yard2
28	50	HgmY3	Hagen Yard 3
29	60	HgnY4	Hagen Yard 4
30	70	Hlb	Hohenlimburh
31	80	HlbY1	Hohenlimburg Yard1
32	90	IsL	Iserlohn-Letmathe
33	100	Alt	Altena
34	110	WerKw	Wehrdohl-Elverlingen Kraftwerk
35	120	Wdl	Werdohl
36	130	PlbY1	Plettenberg Yard1
37	140	Plb	Plettenberg
38	150	PlbY2	Plettenberg Yard 2
39	160	FntY1	Finnentrop Yard1
40	170	Fnt	Finnentrop
41	180	FntY2	Finnentrop Yard2
42	190	FntEnd	Finnentrop Line end

Order: 0 Abbrev: Tracks: 0

Location:

Figure 7 Completed location list for the RSN route.

The completed list may look like Figure 7, 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 6-6 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.

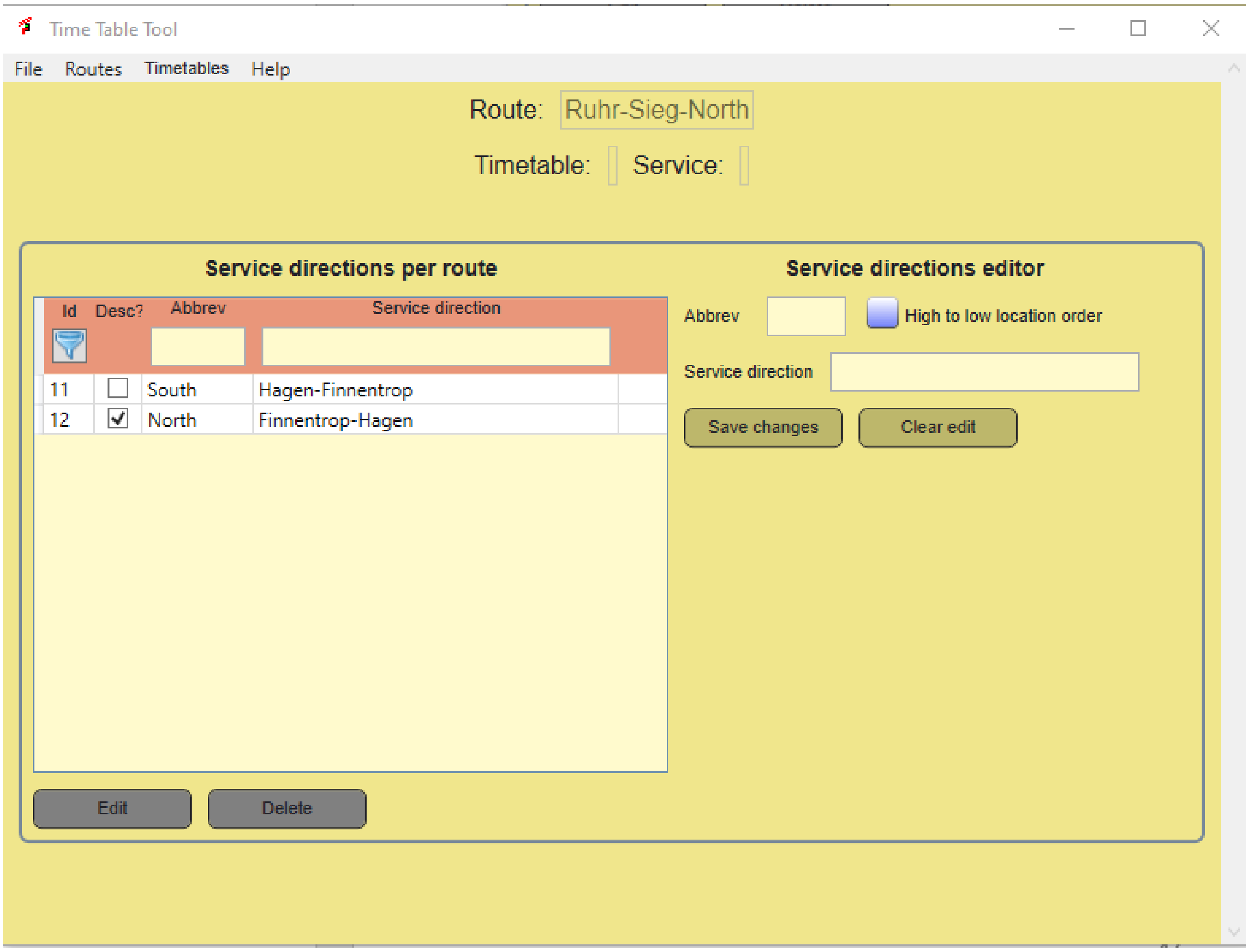
I increase the order by 10, to make it easier to add a location later.

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).



Route: Ruhr-Sieg-North

Timetable: Service:

Id	Desc?	Abbrev	Service direction
11	☐	South	Hagen-Finnentrop
12	☑	North	Finnentrop-Hagen

Figure 8 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.

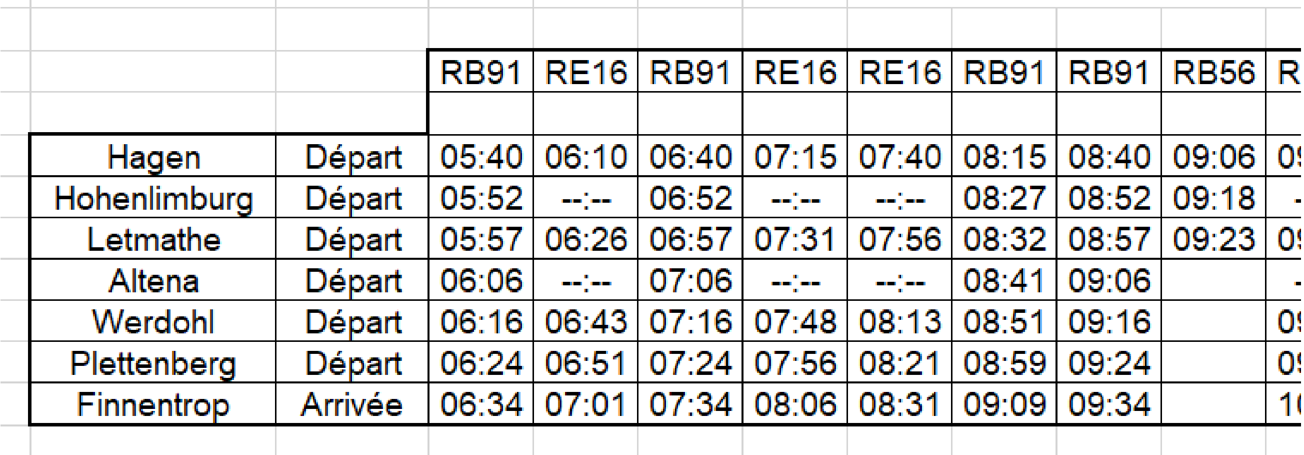
5.4 Services

I order to add services, you must have selected a route. The, at the Timetable menu you can click at the services menu item. Do NOT confuse this with the services instance menu item!. The services page allows you to define a set of re-useable services.

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/

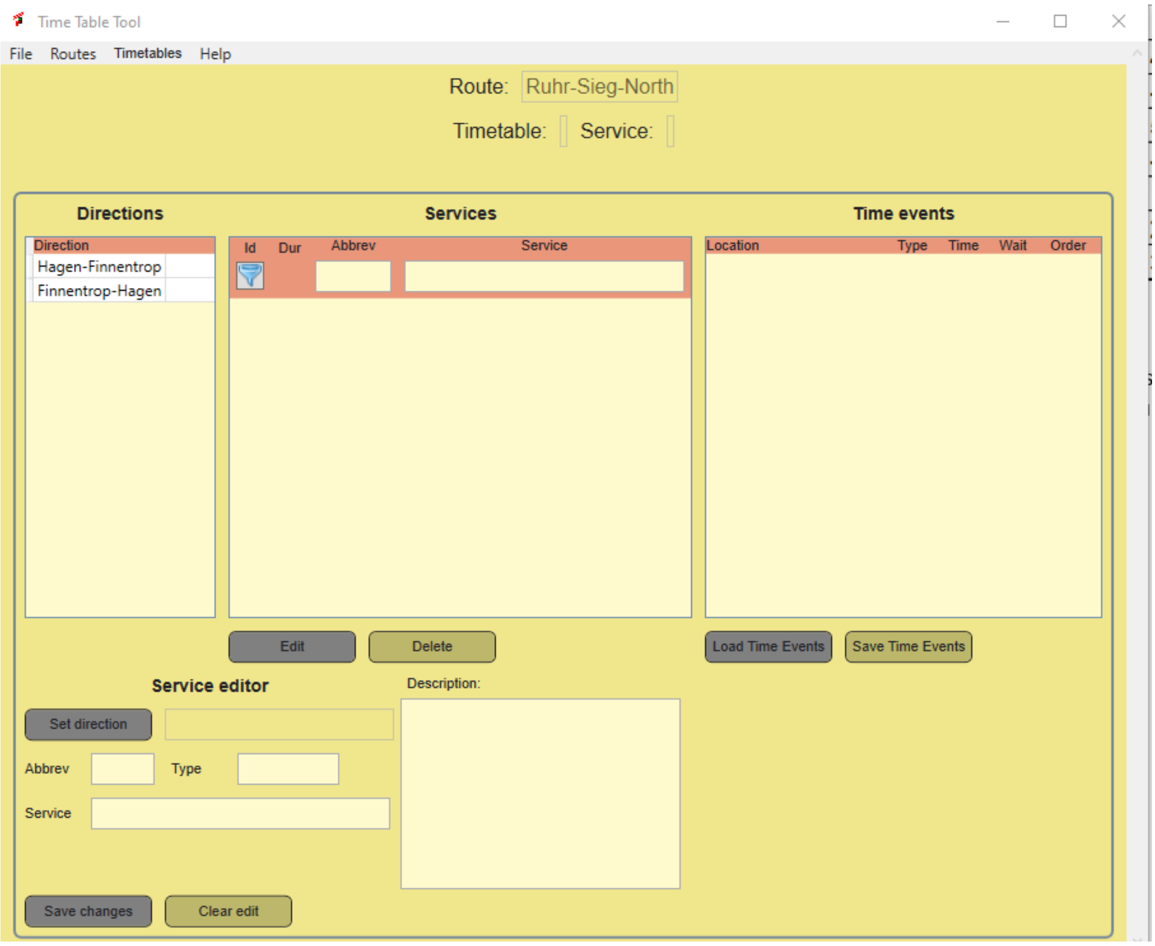
We will create 3 services in each direction, but note they can be re-used.



		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 10 Part of the North-South passenger timetable for RSN

When you have look at this timetable, You see there actually are just three 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 much easier to create these service, as you will see soon.



Route: Ruhr-Sieg-North

Hagen-Finnentrop
Finnentrop-Hagen

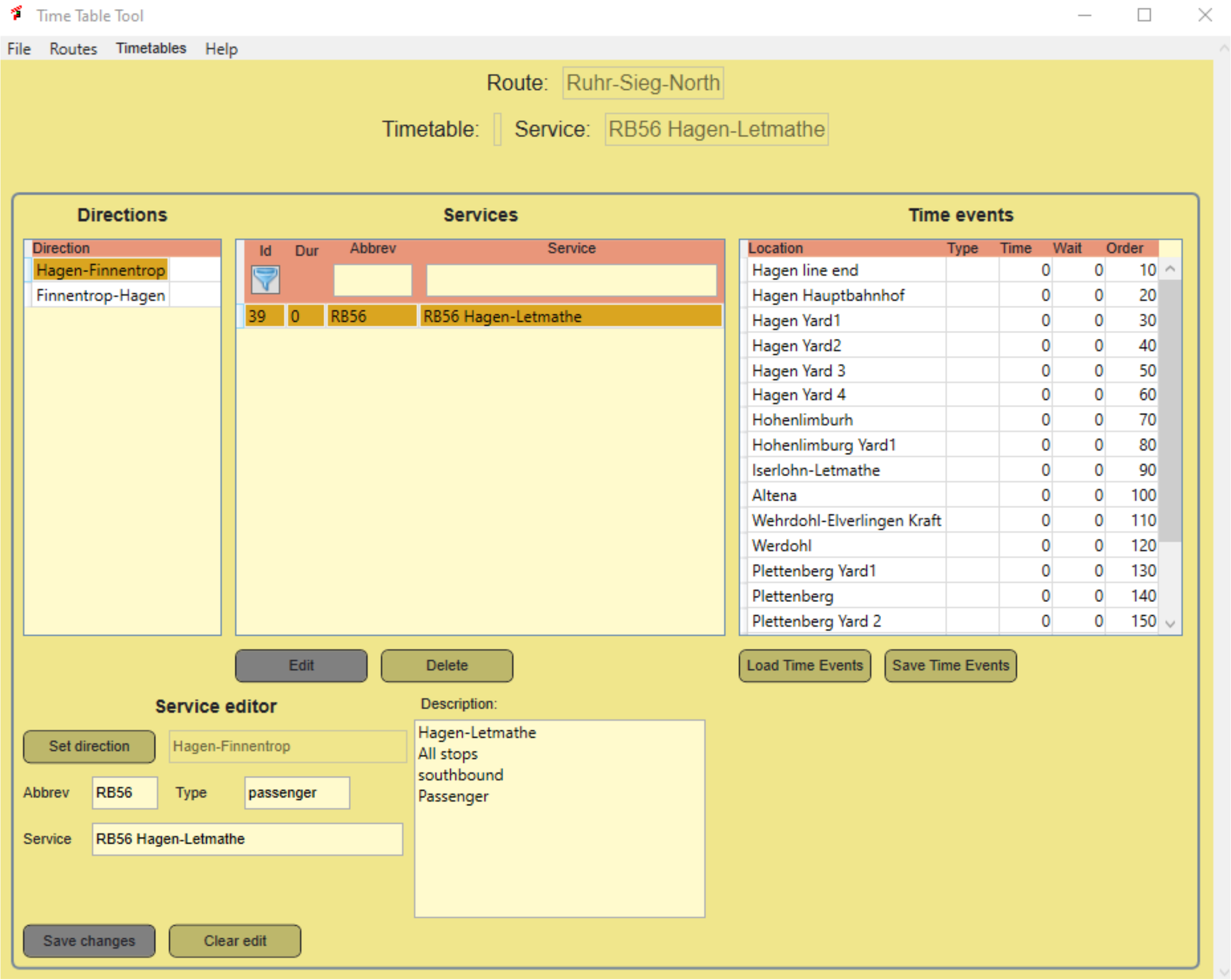
Figure 9 Services form

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.



Route: Ruhr-Sieg-North

Timetable: Service: RB56 Hagen-Letmathe

Direction
Hagen-Finnentrop
Finnentrop-Hagen

Id	Dur	Abbrev	Service
39	0	RB56	RB56 Hagen-Letmathe

Location	Type	Time	Wait	Order
Hagen line end		0	0	10
Hagen Hauptbahnhof		0	0	20
Hagen Yard1		0	0	30
Hagen Yard2		0	0	40
Hagen Yard 3		0	0	50
Hagen Yard 4		0	0	60
Hohenlimburh		0	0	70
Hohenlimburg Yard1		0	0	80
Iserlohn-Letmathe		0	0	90
Altena		0	0	100
Wehrdohl-Elverlingen Kraft		0	0	110
Werdohl		0	0	120
Plettenberg Yard1		0	0	130
Plettenberg		0	0	140
Plettenberg Yard 2		0	0	150

Set direction: Hagen-Finnentrop

Abbrev: RB56 Type: passenger

Service: RB56 Hagen-Letmathe

Description:
Hagen-Letmathe
All stops
southbound
Passenger

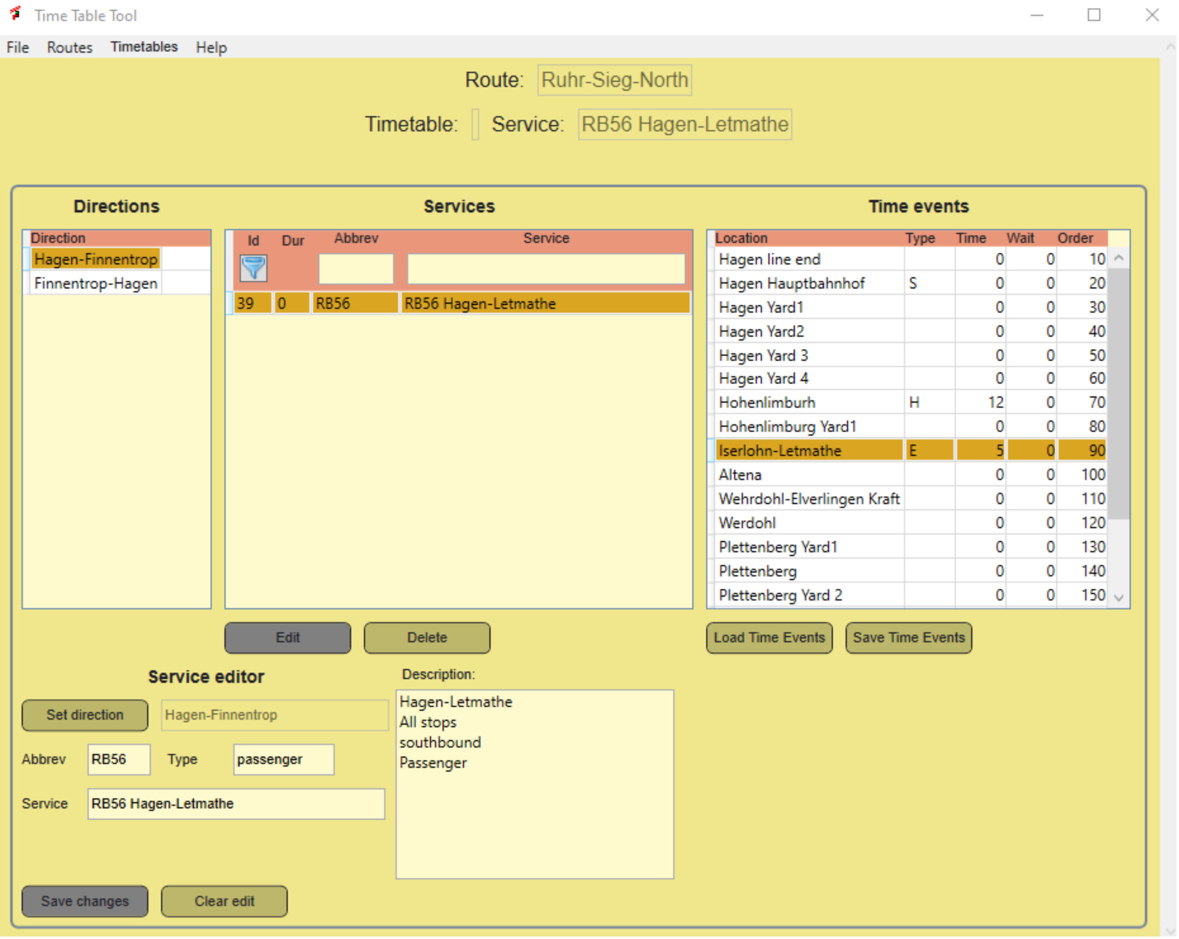
Figure 11 RB56 service defined and time events loaded

In Figure 11 you see the results of the first steps. A service 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 filed type should have the values passenger or freight.

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.
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.

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



Route: Ruhr-Sieg-North

Timetable: Service: RB56 Hagen-Letmathe

Direction
Hagen-Finnentrop
Finnentrop-Hagen

Id	Dur	Abbrev	Service
39	0	RB56	RB56 Hagen-Letmathe

Location	Type	Time	Wait	Order
Hagen line end		0	0	10
Hagen Hauptbahnhof	S	0	0	20
Hagen Yard1		0	0	30
Hagen Yard2		0	0	40
Hagen Yard 3		0	0	50
Hagen Yard 4		0	0	60
Hohenlimburh	H	12	0	70
Hohenlimburg Yard1		0	0	80
Iserlohn-Letmathe	E	5	0	90
Altena		0	0	100
Wehrdohl-Elverlingen Kraft		0	0	110
Werdohl		0	0	120
Plettenberg Yard1		0	0	130
Plettenberg		0	0	140
Plettenberg Yard 2		0	0	150

Set direction: Hagen-Finnentrop

Abbrev: RB56 Type: passenger

Service: RB56 Hagen-Letmathe

Description:
Hagen-Letmathe
All stops
southbound
Passenger

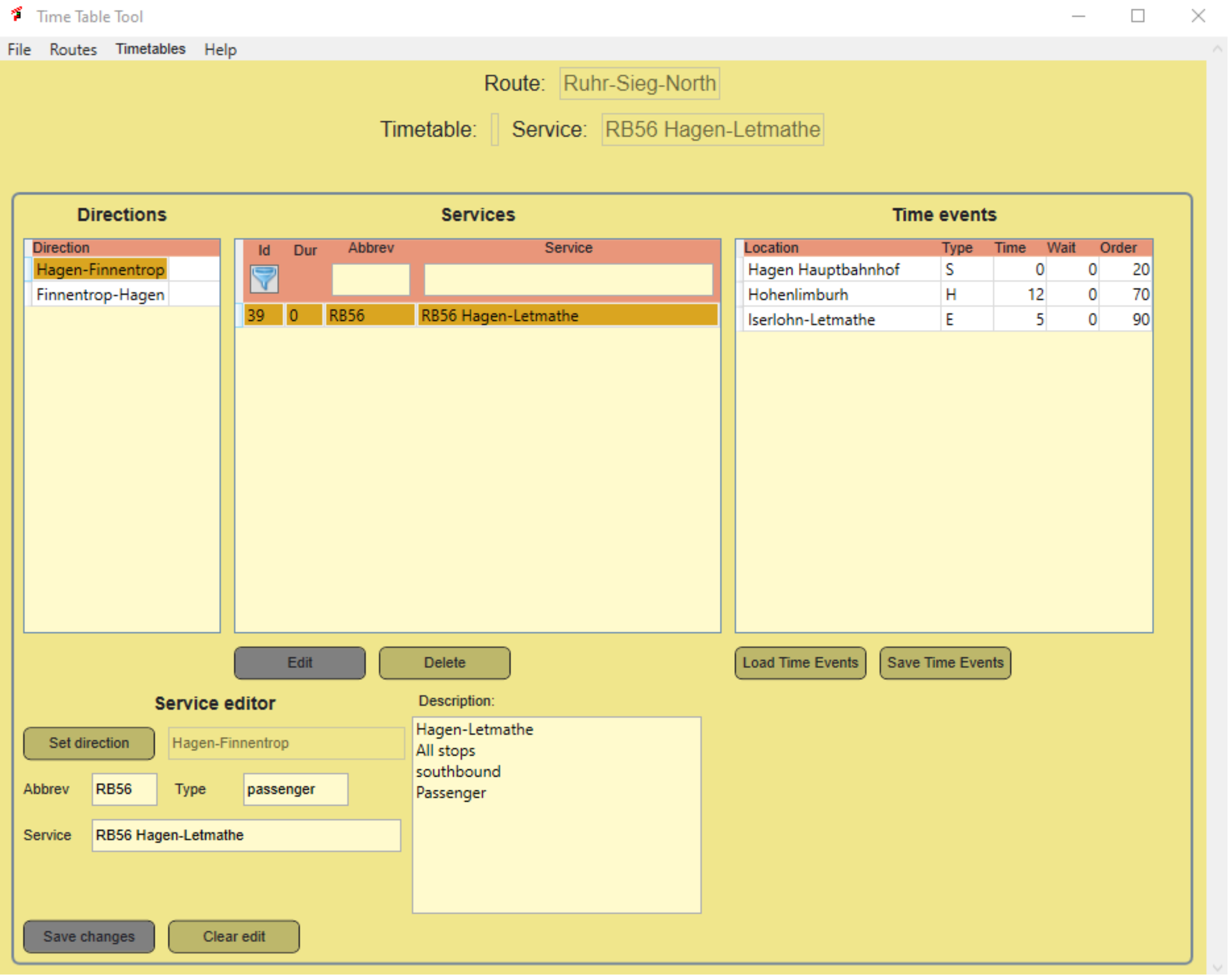
Figure 12 Completed time events for service RB56

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:



Route: Ruhr-Sieg-North

Timetable: Service: RB56 Hagen-Letmathe

Directions

Direction
Hagen-Finnentrop
Finnentrop-Hagen

Services

Id	Dur	Abbrev	Service
39	0	RB56	RB56 Hagen-Letmathe

Time events

Location	Type	Time	Wait	Order
Hagen Hauptbahnhof	S	0	0	20
Hohenlimburh	H	12	0	70
Iserlohn-Letmathe	E	5	0	90

Service editor

Hagen-Finnentrop

Abbrev: RB56 Type: passenger

Service: RB56 Hagen-Letmathe

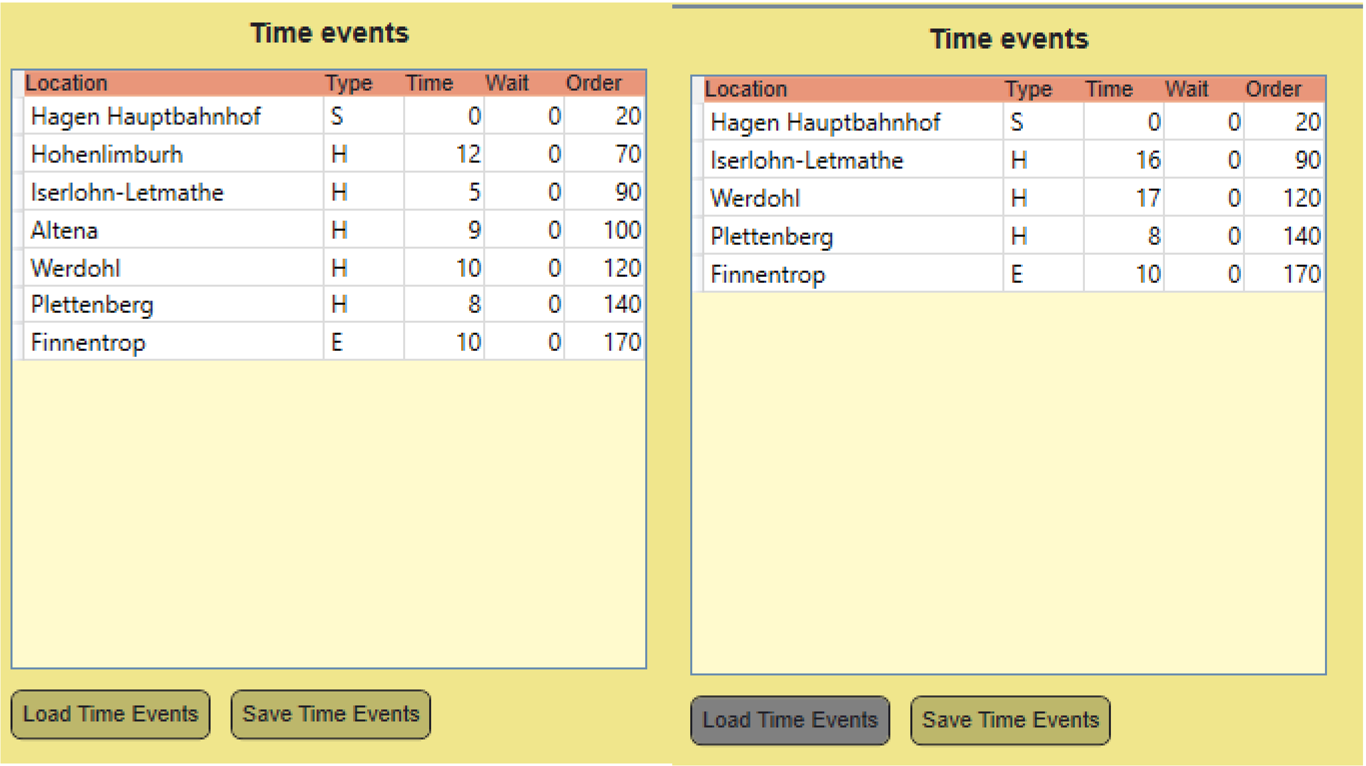
Description:
Hagen-Letmathe
All stops
southbound
Passenger

Figure 13 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:



Time events

Location	Type	Time	Wait	Order
Hagen Hauptbahnhof	S	0	0	20
Hohenlimburh	H	12	0	70
Iserlohn-Letmathe	H	5	0	90
Altena	H	9	0	100
Werdohl	H	10	0	120
Plettenberg	H	8	0	140
Finnentrop	E	10	0	170

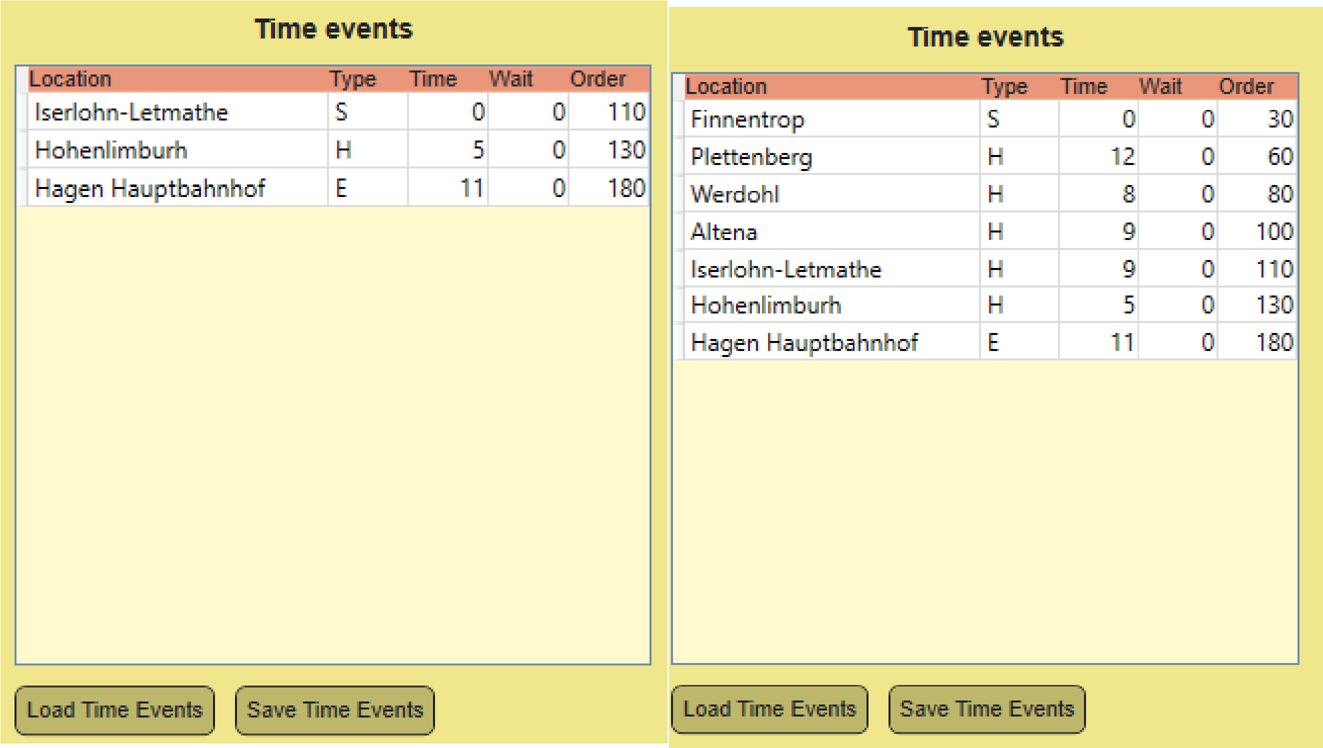
Time events

Location	Type	Time	Wait	Order
Hagen Hauptbahnhof	S	0	0	20
Iserlohn-Letmathe	H	16	0	90
Werdohl	H	17	0	120
Plettenberg	H	8	0	140
Finnentrop	E	10	0	170

Figure 14 RB91 en RE16 services

For the total demo, I will add three services 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.



Time events

Location	Type	Time	Wait	Order
Iserlohn-Letmathe	S	0	0	110
Hohenlimburh	H	5	0	130
Hagen Hauptbahnhof	E	11	0	180

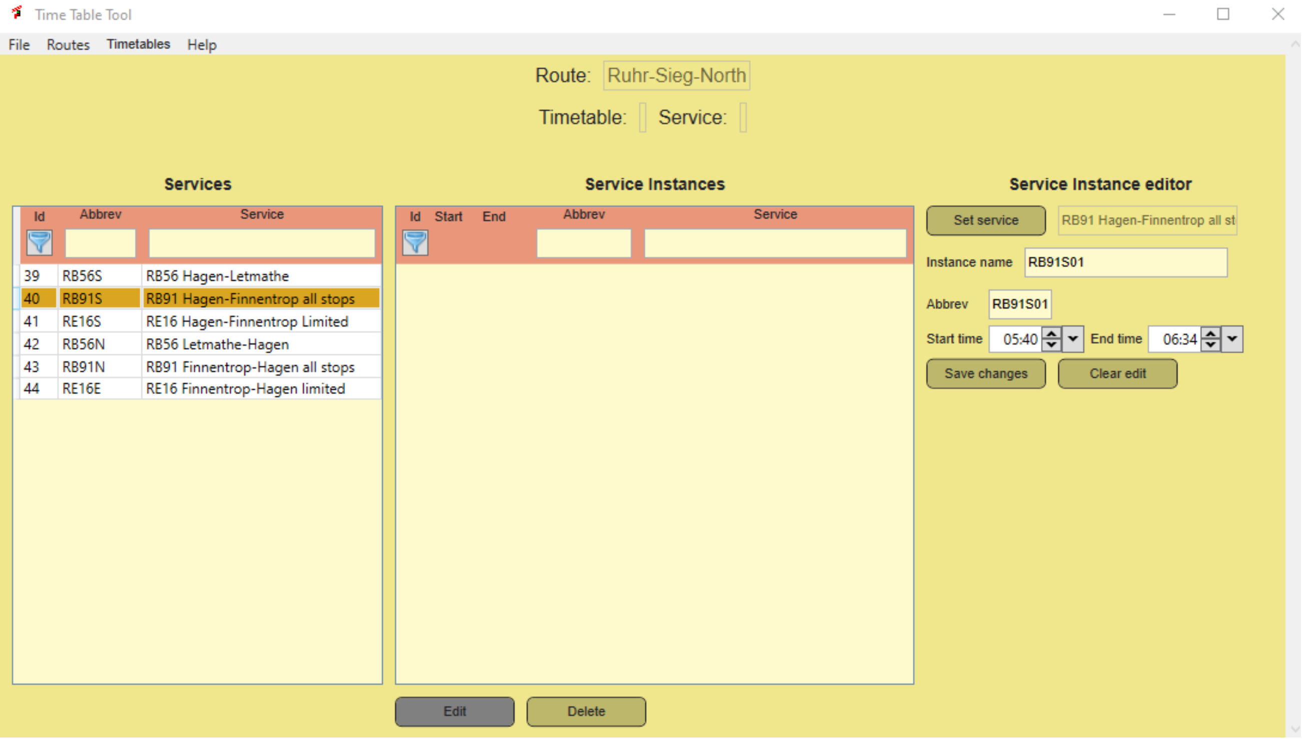
Time events

Location	Type	Time	Wait	Order
Finnentrop	S	0	0	30
Plettenberg	H	12	0	60
Werdohl	H	8	0	80
Altena	H	9	0	100
Iserlohn-Letmathe	H	9	0	110
Hohenlimburh	H	5	0	130
Hagen Hauptbahnhof	E	11	0	180

Figure 15 Two of the three northbound services

5.5 Service instances

Services do not have a start time and they may be repeated several times a day. In this step, start times will be assigned to each service. In the example you see that a number of services are repeated. We will use this to create a number of service instances.



Route: Ruhr-Sieg-North

Timetable:

Service:

Id	Abbrev	Service
39	RB56S	RB56 Hagen-Letmathe
40	RB91S	RB91 Hagen-Finnentrop all stops
41	RE16S	RE16 Hagen-Finnentrop Limited
42	RB56N	RB56 Letmathe-Hagen
43	RB91N	RB91 Finnentrop-Hagen all stops
44	RE16E	RE16 Finnentrop-Hagen limited

Set service: RB91 Hagen-Finnentrop all st

Instance name: RB91S01

Abbrev: RB91S01

Start time: 05:40

End time: 06:34

Figure 16 Define service instances

In order to define service instances, the route must be selected. Once you select Service instances from the menu, the window should look like Figure 16. It has three columns:

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

To create a new service instance, proceed as follows:

1. Select a service in the left column. In the right column, choose this service with the Set Service button.
2. Give the Service Instance a name
3. Give the abbreviation a name
4. 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.
5. Set the end time (this is not yet used)
6. 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 service instances. 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.

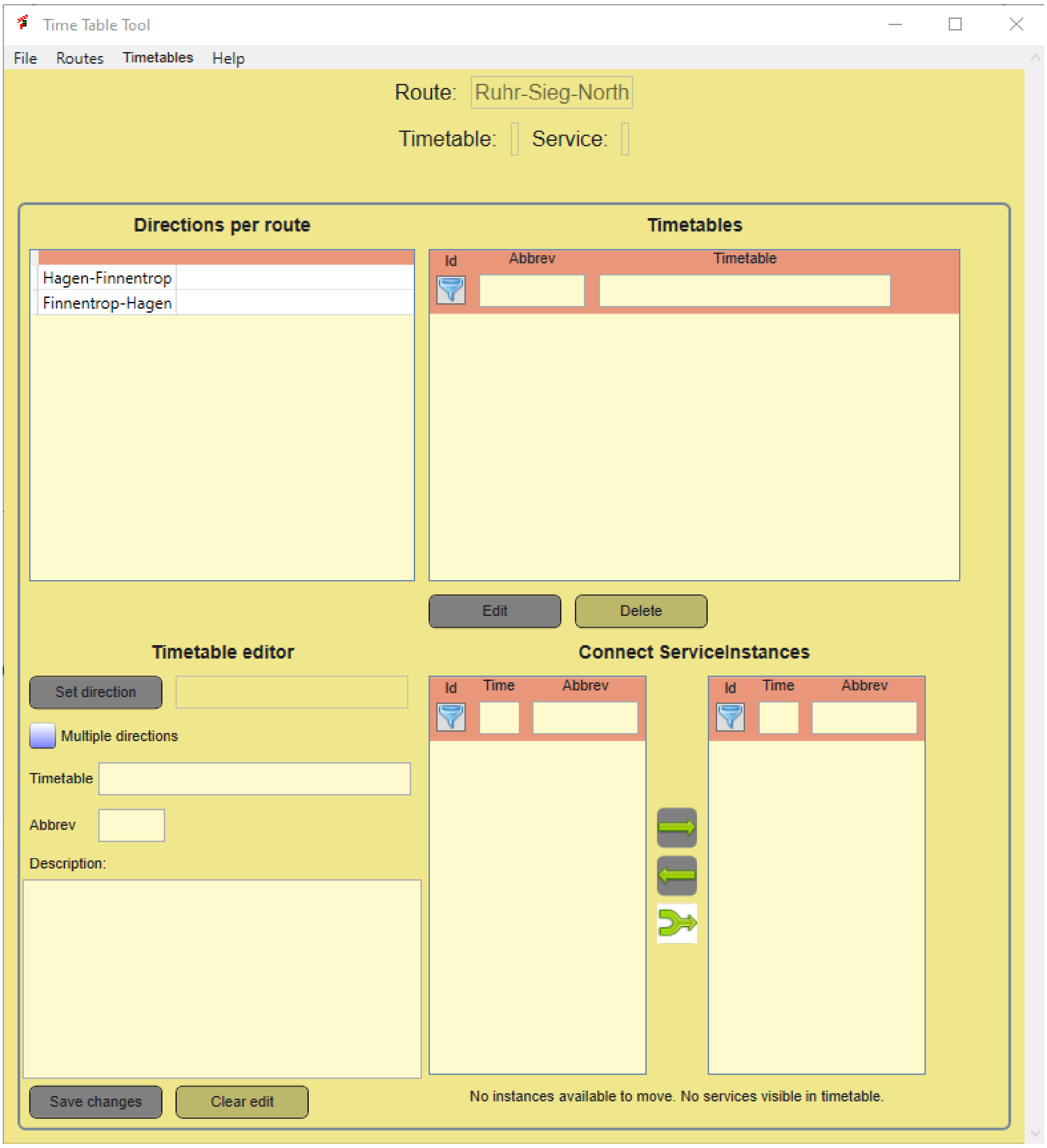
I completed this by adding all services to the service instance list. It still is quite a bit of work, and some interesting shortcuts are on their way for you.

5.6 Timetables

The last step is to add one or more timetables. There a few restrictions. Essentially a timetable is just a set of service instances 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 this:



Route: Ruhr-Sieg-North

Hagen-Finnentrop
Finnentrop-Hagen

No instances available to move. No services visible in timetable.

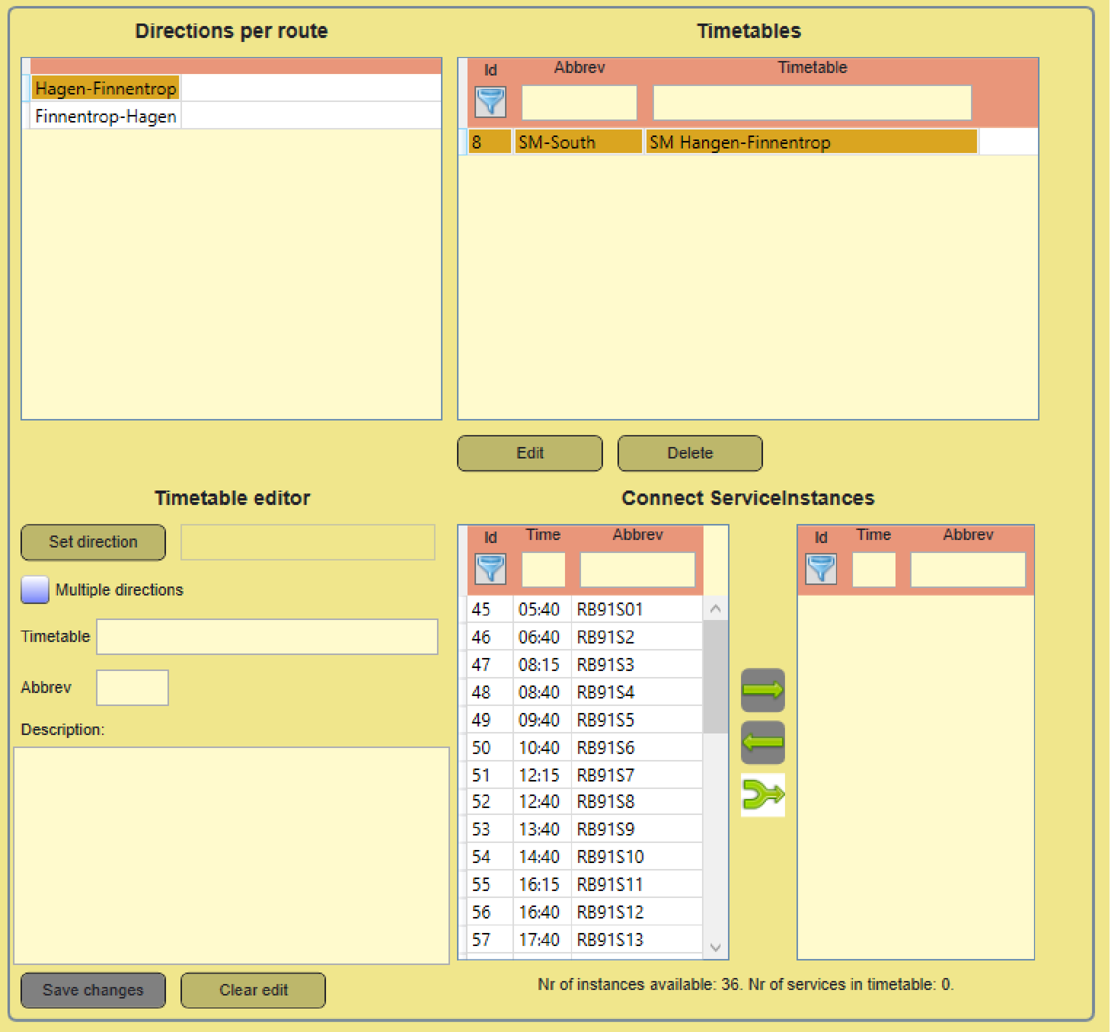
Figure 17 Timetable window

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.



Directions per route

Hagen-Finnentrop
Finnentrop-Hagen

Timetables

Id	Abbrev	Timetable
8	SM-South	SM Hangen-Finnentrop

Timetable editor

□ Multiple directions

Timetable

Abbrev

Description:

Connect ServiceInstances

Id	Time	Abbrev
45	05:40	RB91S01
46	06:40	RB91S2
47	08:15	RB91S3
48	08:40	RB91S4
49	09:40	RB91S5
50	10:40	RB91S6
51	12:15	RB91S7
52	12:40	RB91S8
53	13:40	RB91S9
54	14:40	RB91S10
55	16:15	RB91S11
56	16:40	RB91S12
57	17:40	RB91S13

Id	Time	Abbrev

Nr of instances available: 36. Nr of services in timetable: 0.

Figure 18 Selected timetable showing available service instances

Now you have two options:

- Select a service instance 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 service instances to the timetable at once.

If you want, you can remove a service instance, 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, 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

If you select a route (in the route window) and afterwards a timetable, you can show the timetable you selected. These functions are now in the File menu.

There is one restriction:

You cannot show a timetable with more than one direction in the tabular format. This will be disabled.

6 Reference guide

I assume you read the tutorial, so stuff covered there will not be repeated here. However some advanced stuff is discussed 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 branchlines? It is quite simple, let show that in an example.

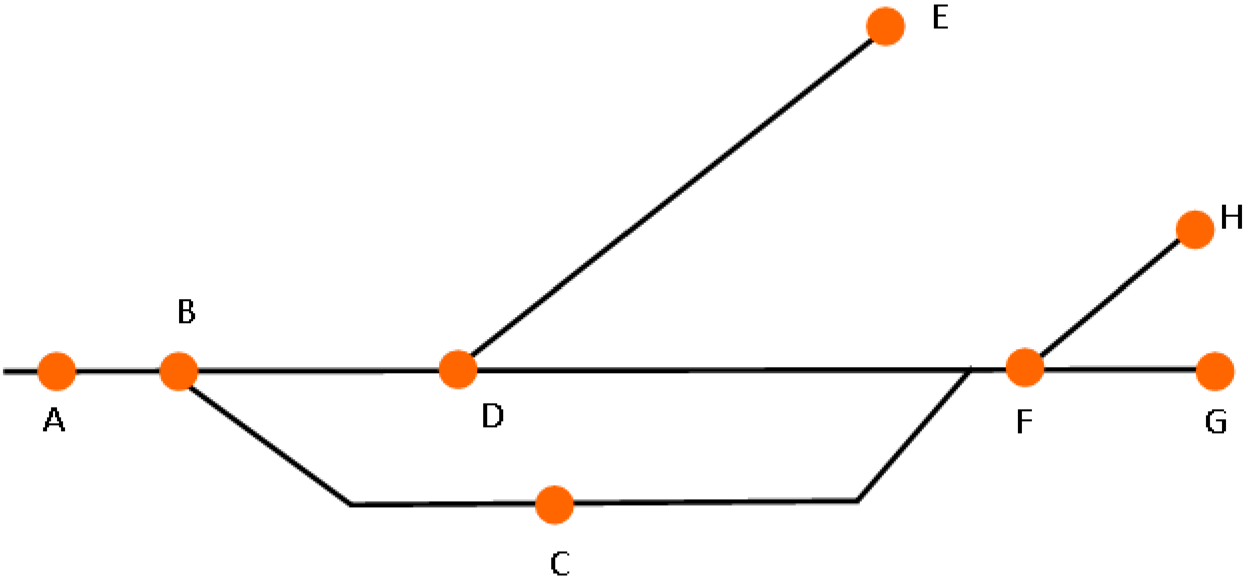


Figure 19 Example of a complex network

In the example as shown in Figure 19 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

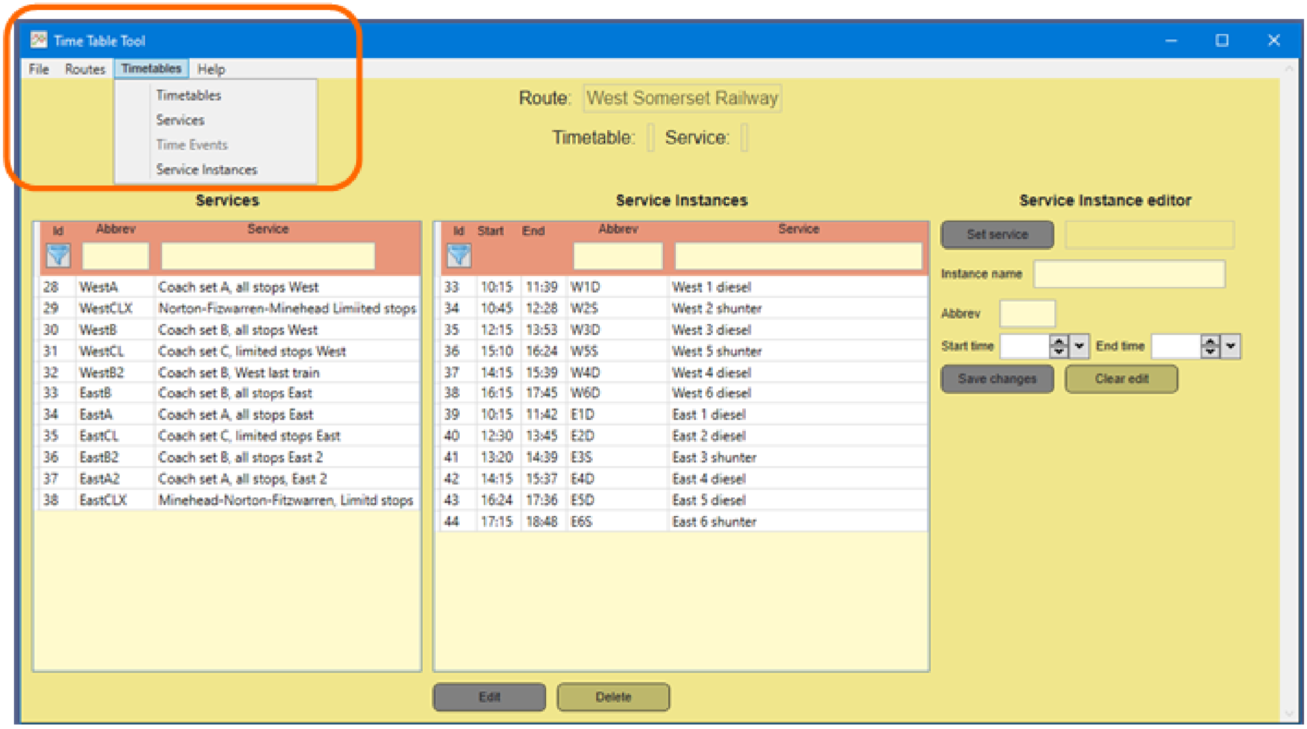
One way to do it, make a diagram and put a ruler on it. If you number strictly from left to right, you will be fine.

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



Route: West Somerset Railway

Services

Id	Abbrev	Service
28	WestA	Coach set A, all stops West
29	WestCLX	Norton-Fizwarren-Minehead Limiited stops
30	WestB	Coach set B, all stops West
31	WestCL	Coach set C, limited stops West
32	WestB2	Coach set B, West last train
33	EastB	Coach set B, all stops East
34	EastA	Coach set A, all stops East
35	EastCL	Coach set C, limited stops East
36	EastB2	Coach set B, all stops East 2
37	EastA2	Coach set A, all stops, East 2
38	EastCLX	Minehead-Norton-Fitzwarren, Limitd stops

Service Instances

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

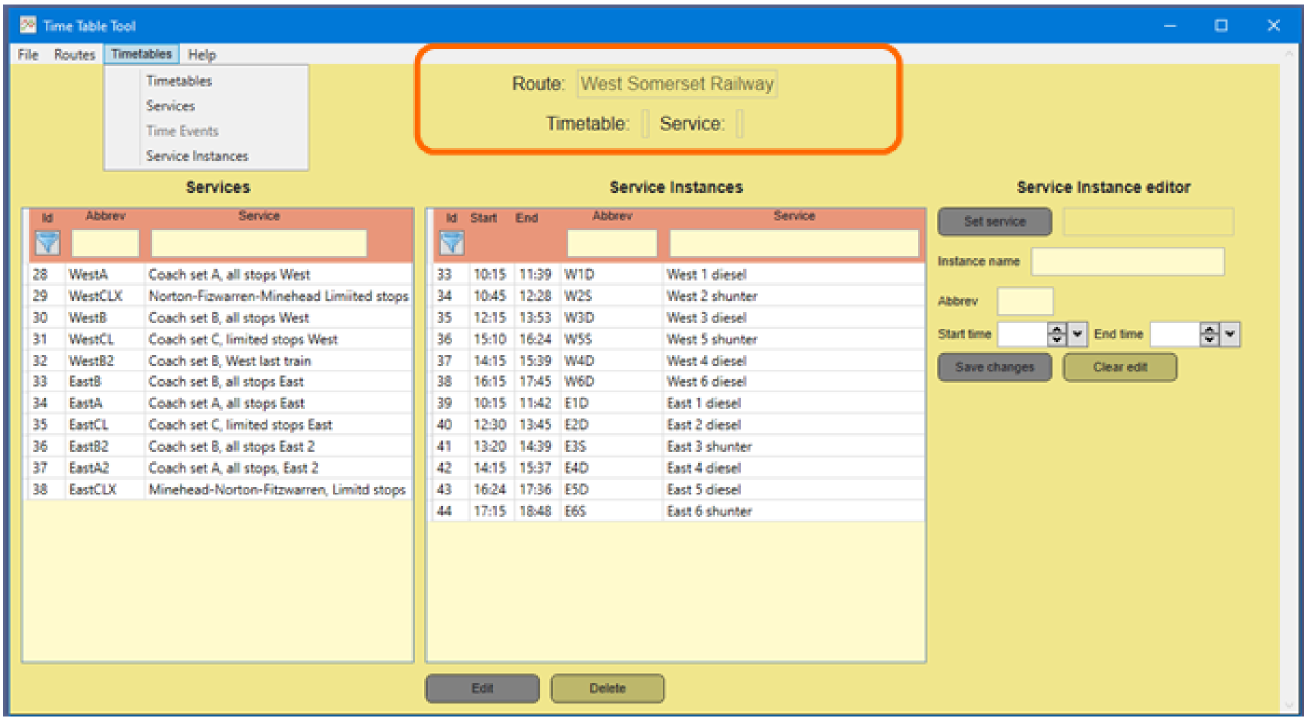
Figure 20 Menu bar

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



Route: West Somerset Railway

Timetable: Service:

Services

Id	Abbrev	Service
28	WestA	Coach set A, all stops West
29	WestCLX	Norton-Fizwarren-Minehead Limiited stops
30	WestB	Coach set B, all stops West
31	WestCL	Coach set C, limited stops West
32	WestB2	Coach set B, West last train
33	EastB	Coach set B, all stops East
34	EastA	Coach set A, all stops East
35	EastCL	Coach set C, limited stops East
36	EastB2	Coach set B, all stops East 2
37	EastA2	Coach set A, all stops, East 2
38	EastCLX	Minehead-Norton-Fitzwarren, Limitd stops

Service Instances

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

Service Instance editor

Instance name

Abbrev

Start time End time

Figure 21 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:

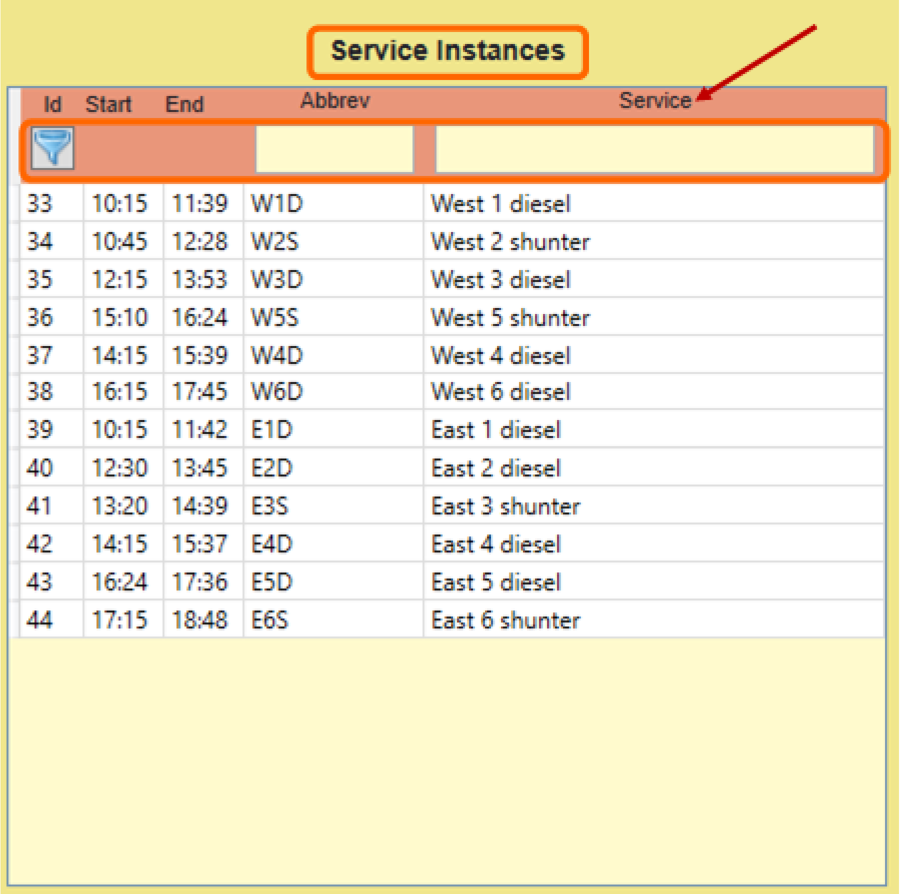
- The selected route
- The selected service
- The selected timetable

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



Service Instances

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 22 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.

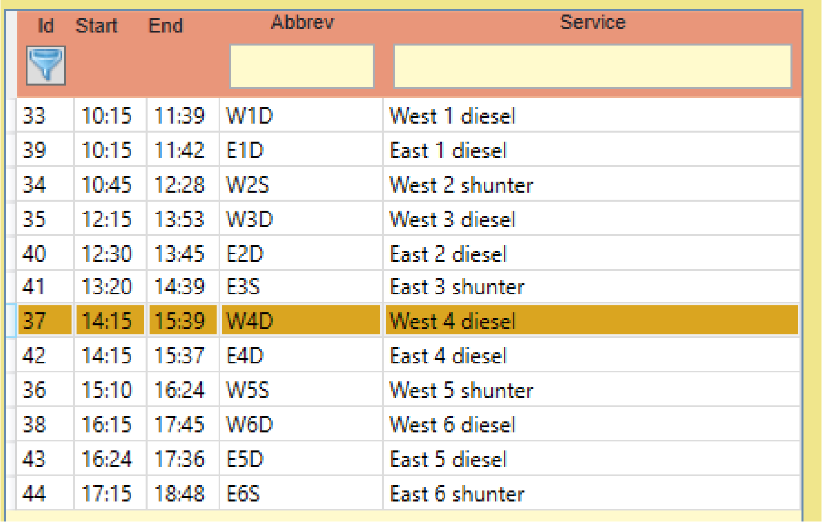
Then, the second marked area, is intended to filter the contents of the table. This not yet doing anything. Worst case it may cause TimetableTool to crash. In future, these filters will be removed or they will be functional.

In between is the headerline. 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 23 Table with selected row

6.2.4 Buttons

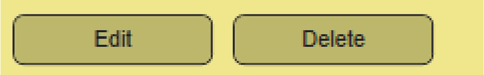


Figure 24 Buttons below a table

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. This NOT yet implemented, so the button is not functional at the moment.

6.2.5 Editor

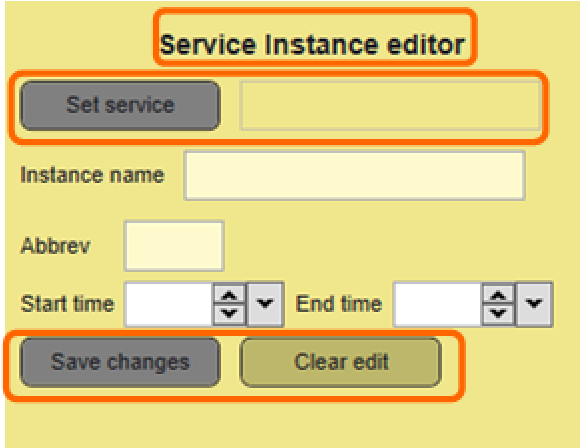


Figure 25 Editor part

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.

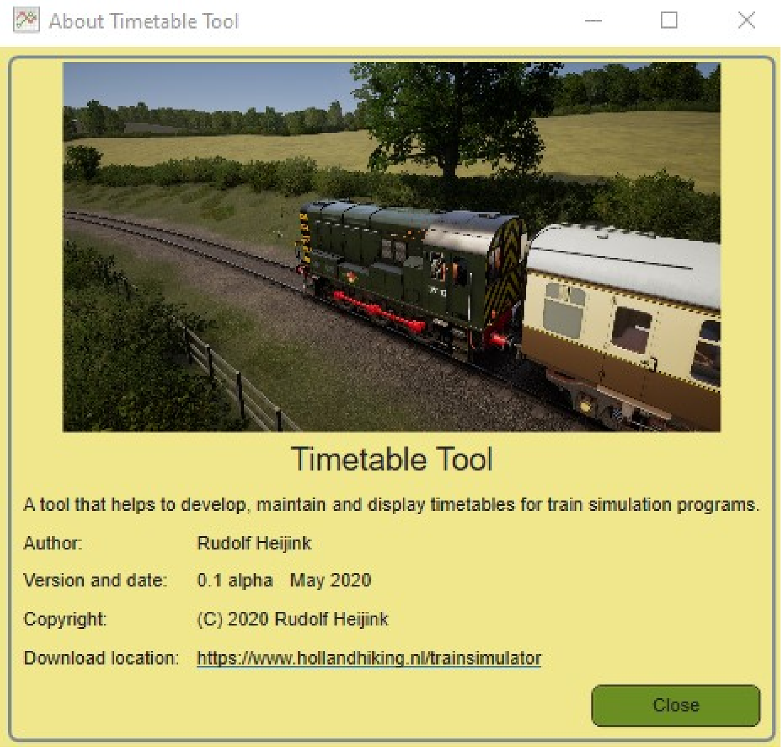
At the second line of the edit tool, you may see a button labelled Set …. (Set service in the example). This button will be activated if you select a row in the left most column. Then you should press the button and in the field next to the button, your selection will appear. In this way you connect 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 preset time with accuracy of 30 minutes. I do not like this control very much and it definitely will be replaced later this year.

6.3 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.



About Timetable Tool

Timetable Tool

A tool that helps to develop, maintain and display timetables for train simulation programs.

Author: Rudolf Heijink

Version and date: 0.1 alpha May 2020

Copyright: (C) 2020 Rudolf Heijink

Download location: https://www.hollandhiking.nl/trainsimulator

Figure 26 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 …

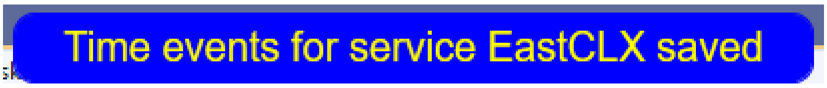
You must close this window before you can continue.

6.3.1 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.

6.3.2 Logging

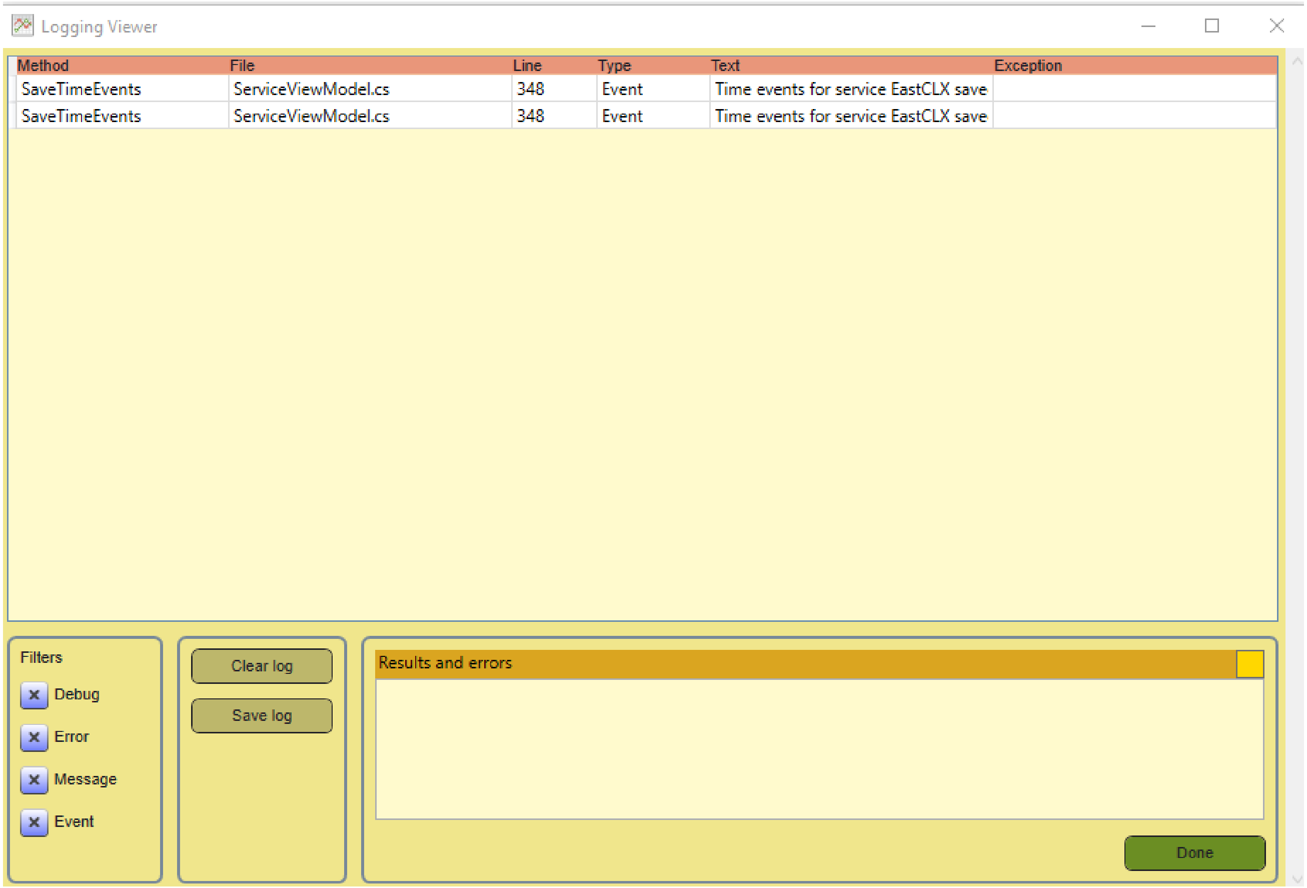
The last item in the help menu is the Logging. If you click, a window is opened that shows all log messages. A log message is created in most cases if an error occurs, or just to inform you about something that happened. In most case, you also will see a blue popup screen at the top of your window.



Time events for service EastCLX saved

Figure 27 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.



Method	File	Line	Type	Text	Exception
SaveTimeEvents	ServiceViewModel.cs	348	Event	Time events for service EastCLX save	
SaveTimeEvents	ServiceViewModel.cs	348	Event	Time events for service EastCLX save	

Figure 28 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.

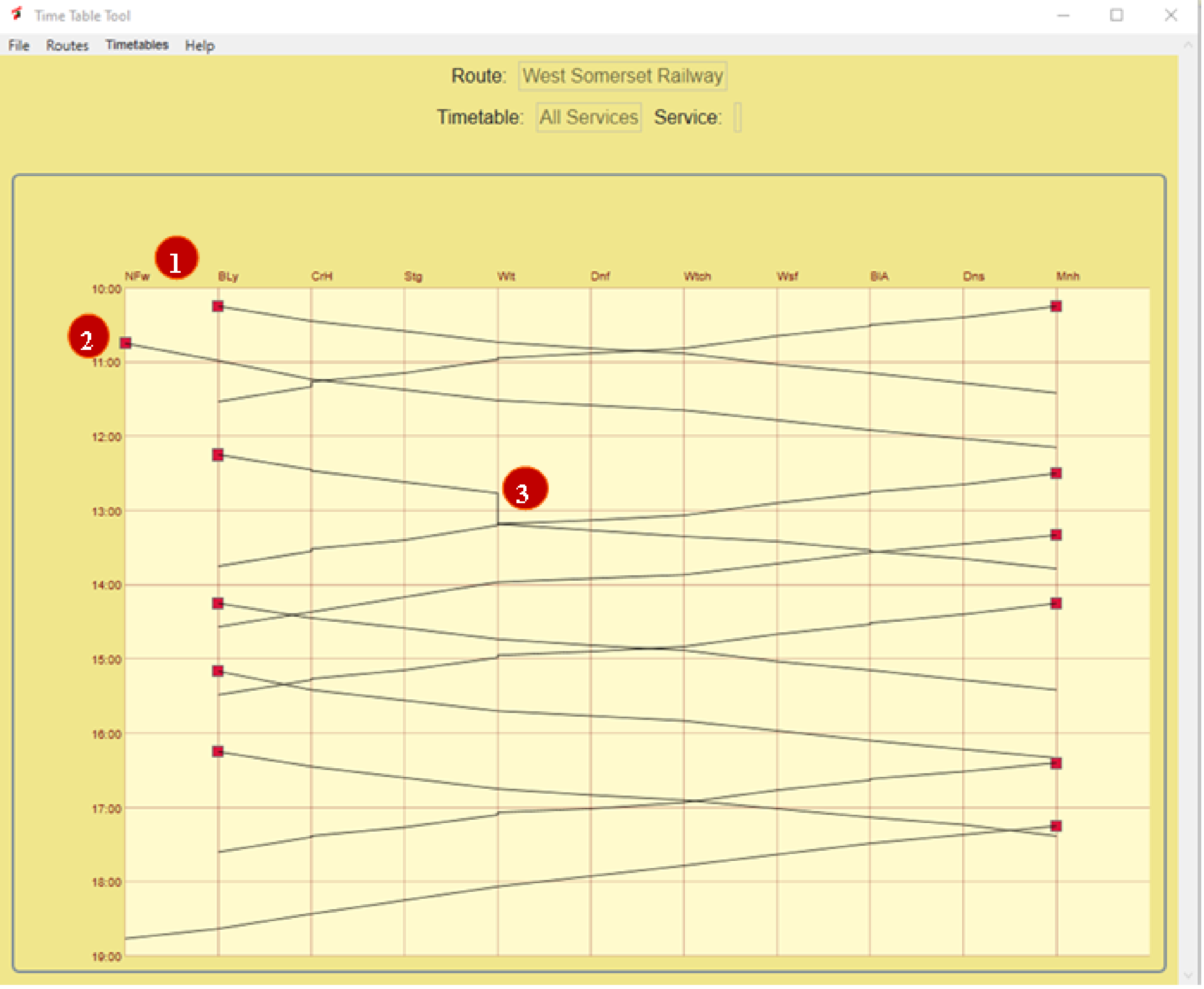
If you experience any problems, please make an export and send it to me. This is very helpful in finding issues.

6.4 Viewing timetables

As stated before, there are two 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 other view, shows a graph. This view can show services in two directions in the same graph.

This is a first version and a lot of improvements will come. I like to point at some details.



Route: West Somerset Railway

Timetable: All Services Service:

Figure 29 Timetable graph

At the horizontal axis you see the location abbreviations. If you move the mouse cursor over them, a tooltip will show the full name

The red dots are buttons, located at the start of a service. Here also a tooltip will show some more details. In future, here a number of additional actions may be added.

If you see a vertical line, a train is waiting here. You can see this enables trains to pass each other. For the first service you can see right away that this timetable is not correct. Trains cannot meet each other outside the stations.

6.5 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:

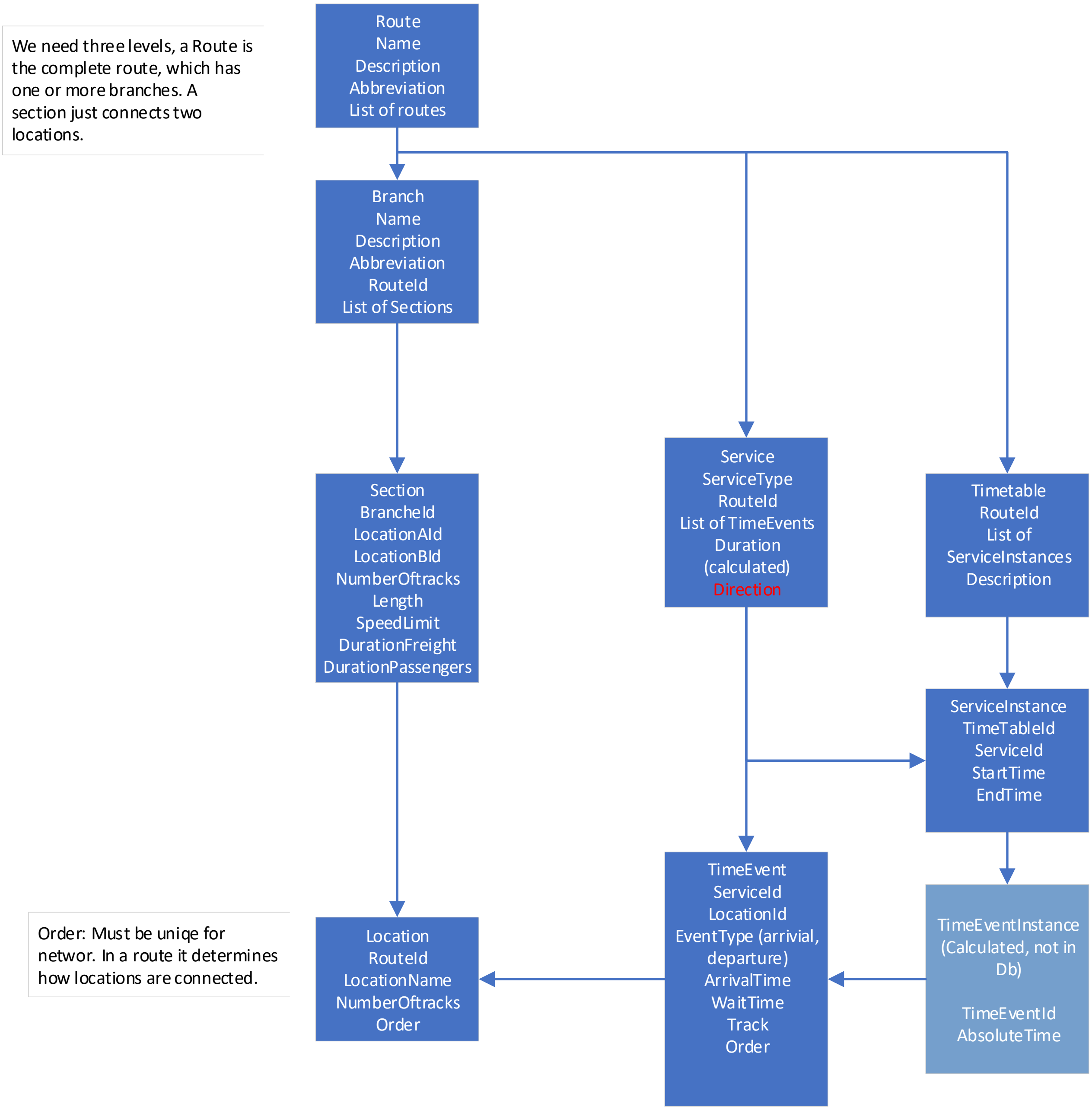


Figure 30 Database structure

There is one tricky point. Because service instances 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).

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

6.5.1 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. SQlLite

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

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	Delete buttons not working	high

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