

User Manual
TransportControl v3.1



Table of contents

1.	Introduction	1
2.	Interface and usage	1
2.1.	Browser UI call and log in	1
2.2.	User interface Main UI	1
2.3.	User interface Custom UI	2
2.4.	Tools	2
2.4.1.	Zoom in/Zoom out	2
2.4.2.	Turn map clockwise/counterclockwise	2
2.4.3.	Reset map view	2
2.4.4.	Zoom to selected element	2
2.4.5.	Show envelopes	2
2.4.6.	Selection frame	2
2.4.7.	Job-Quick-Placement	2
2.4.8.	Draw segments and contours	3
2.4.9.	Undo last drawing action	3
2.4.10.	FN	3
2.5.	Resource table operation	3
2.6.	Up-/Download	3
3.	Targets	4
3.1.	Real Targets	4
3.2.	Simulate targets	4
3.3.	Resource table Targets	5
3.4.	Control targets	5
3.5.	Traffic management	6
4.	Segments	7
5.	Jobs	11
6.	Contours	14
7.	Events	16
8.	Variables	18
9.	Scripts	18
9.1.	Typ SERVER	21
9.2.	Typ UPDATE	21
9.3.	Typ Custom	21

9.4.	Typ INPUT und OUTPUT	21
10.	Models	21
10.1.	General	21
10.2.	Layout	22
10.3.	Shape	23
10.3.1.	Tools	24
10.3.2.	Adjustment of drawn shapes	24
10.4.	Status	24
10.5.	Errors	25
10.6.	Actions	26
10.7.	Properties	26
11.	Settings	27
12.	Users	27
12.1.	Change password	29
13.	Logfiles	30
14.	PANIC	31
15.	Account	31

Table of figures

Figure 1: Toolbar	2
Figure 2: An empty resource table	3
Figure 3: Ressource table Targets	5
Figure 4: Remote control	6
Figure 5: Dynamic traffic control through envelope curves	7
Figure 6: Resource table Segments	7
Figure 7: Segments Editor – General	8
Figure 8: Segment Editor - Properties	9
Figure 9: Ressource table jobs	11
Figure 10: Job Editor - General	11
Figure 11: Job Editor - Orders	12
Figure 12: Job Editor - Properties	13
Figure 13: Set contours points	14
Figure 14: Draw contour	14
Figure 15: Contour Editor	15
Figure 16: Resource table Contours	16
Figure 17: Resource table events	16
Figure 18: Event editor – General	17
Figure 19: Event Editor - Tasks	17
Figure 20: Resource table variables	18
Figure 21: Variable Editor – General	18
Figure 22: Resource table scripts	19
Figure 23: Resource table scripts encryption	19
Figure 24: Resource table scripts – General	20
Figure 25: Script Editor – Code	21
Figure 26: Model Editor – General	22
Figure 27: Model Editor – Layout	22
Figure 28: Model Editor – Shape	23
Figure 29: Toolbar of the shape editor	24
Figure 30: Model Editor – Status	25
Figure 31: Model Editor - Errors	26
Figure 32: Model Editor – Actions	26
Figure 33: Model Editor – Properties	26
Figure 34: Resource table Settings	27
Figure 35: Resource table users	28
Figure 36: User Editor – General	28
Figure 37: User Editor – Permissions	29
Figure 38: Account window	29
Figure 39: Resource table Logfiles	30

List of tables

Table 1: Segment Properties, presentation and effects	10
Table 2: Model properties, value entries and effects	27
Table 3: Log level scope	30



Glossary

• Resources	=	All usable objects that can be changed at runtime (vehicles, segments, etc.)
• Browser UI	=	User Interface of TC; a distinction is made between Main UI and Custom UI
• Main UI	=	Standard user interface of TC
• Custom UI	=	Customized user interface
• Main Menu	=	Main menu; located at the top of the screen
• Toolbar	=	Control bar, which is located on the left edge of the screen
• PANIC-Button	=	Emergency button in the Main Menu marked "PANIC"
• 2-Finger-Pinch	=	Gesture function for touch screens with two fingers
• Core loop	=	Loop in which the various resources within TransportControl are called and processed.
• Tools	=	Tools that can be selected via the toolbar
• Envelopes	=	A curve that touches each share curve of a curve group once; used in TransportControl for collision avoidance.
• Targets	=	Network participants that communicate with TransportControl via the UDP interface
• UDP- Protocol	=	User Datagram Protocol for the transmission of Information between Target and TransportControl
• Target Key	=	Eight-digit code for unique vehicle identification and definition of properties for the target
• Orders	=	Subtasks of a job or transport order
• TO	=	Transport order
• Segments	=	Virtual route sections within TransportControl
• ID	=	Naming for unique identification of a resource; Is used to call the resource within the different functions of TC

1. Introduction

This document describes the usage of TransportControl. The handbook refers to the use of the management control software with the supplied graphical user interface. The chapters of the document deal with the general operation and the administration of the individual resource groups.

2. Interface and usage

2.1. Browser UI call and log in

The requirement for calling the Browser UI is a current browser with HTML5 support.

The Main UI is the standard user interface of TransportControl in which the full range of features can be used. A user interface tailored specifically to the customer's requirements is called Custom UI.

The Browser UI is called up by entering one of the following addresses in the address line of the browser:

<code>http://[SERVERIP]:[SERVERPORT]/transportcontrol</code>	Main UI
<code>http://[SERVERIP]:[SERVERPORT]/transportcontrol/?script=[UINAME]</code>	Custom UI

During the initial registration, a request appears to download the license via a link from a server of movizon and to drag the license file into a field provided for this purpose. For this purpose, use the username to that the license is registered.

For more information about the installation process and system requirements, please use our separate installation guide, which you can find on our homepage.

After the software is licensed, a simple authentication is all that is needed to log into the system.

If a direct login with a specific user is requested, this is also possible. For this purpose, the access data must be appended to the link to the Browser UI as follows:

<code>http://[SERVERIP]:[SERVERPORT]/transportcontrol/?username=[USERNAME]&password=[PASSWORD]</code>
<code>http://[SERVERIP]:[SERVERPORT]/transportcontrol/?script=[UINAME]&username=[USERNAME]&password=[PASSWORD]</code>

2.2. User interface Main UI

The user interface is divided into main menu, map, toolbar and info bar.

In the upper display area, there is the main menu with the buttons for the 10 resources as well as the log files, the **PANIC Button** and the user administration. Every user is only shown the resources that he has been authorized to view.

It is possible to open several resource tables by clicking the corresponding buttons and to move these tables independently. The resource tables **Segments** and **Contours** are an exception. These cannot be opened in parallel.

The map is based on a two-dimensional coordinate system with a grid. It is used to visualize the segments, contours and vehicles. The origin of the coordinate system is marked with a rhombus.

In the center of the left side of the screen is the toolbar. The tools allow interaction with the map and the resource tables. The exact functions are described in section 2.4 Tools.




The currently used TransportControl version, the logged-in user, and the dimension of the grid on the map can be found left-aligned in the info bar at the bottom of the screen. If the mouse pointer is on the map, the exact position of the mouse pointer is also displayed on the left side with X and Y coordinates. Information such as the date, the current server time, the working clock, and the current core loop interval time are displayed on the right side. The core loop interval time reflects the time period that passes during resource management and is therefore a measure of the server's utilization.



2.3. User interface Custom UI

A **Custom UI** is used to provide a simplified user interface that has been individually tailored to the use case. The basis of every **Custom UI** is the header and footer. These cannot be changed. The **PANIC Button** and the account settings are available in the header. In the footer, the current TransportControl version, as well as the date and the current server time can be read. The area between header and footer is configured by a specific script.



2.4. Tools




2.4.1. Zoom in/Zoom out

 and  are used to zoom. This function is needed, for example, for laptops without a mouse or touch screens without multi-touch.


2.4.2. Turn map clockwise/counterclockwise

With  and  the coordinate system of the map is rotated either left or right by 90°.


2.4.3. Reset map view

When pressing , the zoom and position is adjusted so that all contours and segments are visible on the screen at the same time. After logging in to the Main UI, this view is always displayed.

2.4.4. Zoom to selected element

After selecting an object in one of the resource tables, clicking on  focuses the view of the map on this object. This affects zoom and position on the map.

2.4.5. Show envelopes

Via  the display of the simulated envelopes can be switched on and off in TransportControl.

2.4.6. Selection frame



By activating the  button, several objects can be captured on the map at the same time by drawing a frame around the elements to be selected. For this purpose, the corresponding resource table must be open.



Figure 1: Toolbar

2.4.7. Job-Quick-Placement


After activating  it is possible to drag and drop vehicles once with a job to the target segment.





2.4.8. Draw segments and contours

 is the drawing tool for segments.  is the drawing tool for contours.

2.4.9. Undo last drawing action


Via , the last drawing action (segment/ contour/ contour point) can be undone.

2.4.10. FN

With activated  the entries can be selected in the tables from the first click to the second click. A selection of several objects on the map and their individual deselection is also possible. In addition, straight lines can be forced when drawing segments while this function is activated. The "Shift" key activates the function of  as long as it is held down.

2.5. Resource table operation

All resource tables have a similar structure. In the upper left corner is the name of the resource table, oriented to the right are the editing fields and below that the list of the related elements in tabular form. If a resource or multiple resources are selected on the map with an open resource table, they will also be selected in the table.

Via , all elements in the resource table can be selected. By pressing , all selected elements are locked. Once an element is locked, this element can only be unlocked again by the same user.

The further editing fields provide different editing functions for one or more elements. Their functions are explained in the corresponding section for the respective resource group.

There are tooltips for the individual icons in the resource tables. These describe the function of the individual elements. To display them, move the mouse over the respective element without clicking on it. Then a short description is displayed.

Via the **Search** field, you can enter specific terms. If the resource table contains entries with this term, these jobs will be displayed first in the list.

Jobs

Figure 2: An empty resource table

2.6. Up-/Download

For easy exchange of TransportControl data, the Upload and Download functions have been implemented. An exception exists for the resource group Targets, for which no upload or download function is provided, and for the log files, which can only be downloaded.

For the resource groups jobs, user, events, variables, segments and contours, all data of this resource group is affected by an upload or download. In the resource table scripts and models, the function only affects a single resource. For the combination of different segment or contour files there is exclusively for these two resource groups the additional function Merge, within the upload window. If this function is activated before confirming an upload, only existing IDs will be overwritten. The data downloaded from the TransportControl database is converted into the JavaScript Object Notation (JSON) format.

The log files can be downloaded by all users.



If, when uploading segment, contour, model or script files, the file is corrupted or incorrect the Bad Request message is displayed when you confirm the upload. Errors can also be recognized in the case of segments and contours by the fact that the imported files are not displayed graphically in the upload window, but in text form.

3. Targets

Targets are all participants that communicate with TransportControl via UDP telegrams. Targets can be, for example, vehicles, traffic lights, gates, sensors or conveyor belts. Each target has its own ID. The ID is an eight-digit unique identifier and consists of "TCV" ("TransportControl Vehicle") for vehicles or "TCS" ("TransportControl Station") for stations and a freely selectable five-digit string. The assignment to a specific vehicle class with defined properties such as appearance, actions, characteristics, behavior, etc. is done via the target key, which directly follows the specification "TCV" or "TCS". The target key is defined in the model and can be taken from the resource table of the same name. The document "TransportControl Target API" contains further explanations on the integration of targets. It also explains how to define a communication protocol and how to enter status and error messages.

3.1. Real Targets

A real target logs on to TransportControl when the first UDP telegram is received. Deleting the target results in the message "Access denied" if the vehicle is not offline. A target is considered offline if TransportControl does not receive any UDP protocols from a target for a certain period of time. The length of this period depends on the settings of the respective model. Vehicles that are offline will continue to be displayed in TransportControl at the last known position.

Because TransportControl cannot determine whether the vehicle is still at this position, the envelope curve in TransportControl continues to be taken into account. Further FTFs would therefore stop to avoid a collision.

If the vehicle is physically no longer at this position, it can be completely removed via the **Remove** button in the Targets resource table. This has the effect that the Mileage attribute of the vehicle is reset to zero.

The information of the telegrams sent to TransportControl can be read out by pressing the I/O button after previously selecting a target in the resource table. This opens a window that displays the transmitted telegrams in real time. This function is used for troubleshooting and analysis and is usually used by developers or FTF manufacturers.

3.2. Simulate targets

To simulate a target, the Targets resource table must be opened and the **Simulate** button must be pressed. After that, the ID and the location where the simulated vehicle should be created must be selected. Removing a simulated target is possible via **Remove**. A dialog box opens with a warning, which must be confirmed. This now causes a connection break between TransportControl and the simulated vehicle. Depending on the length of the defined offline period, the simulated target can then be finally removed a short time later by selecting the **Remove** button again. After a restart of the server, TransportControl automatically regenerates all simulated targets.









3.3. Resource table Targets

All targets - whether simulated or real - are listed in the Targets resource table. There, the ID, status information of the target and information of the transport requests are listed.



Targets										Search	I/O	Update							
		ID					Response	Update	Mileage [km]	Location	Destination	Modified		Responsibility					
<input type="checkbox"/>		TCVWSL_2					205 ms	NONE	0.00	ID 57	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVWSL_1					206 ms	NONE	9.72	N8	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVP60_2					204 ms	NONE	0.01	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVP60_1					204 ms	NONE	0.00	S1	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCBR0001					204 ms	NONE	0.00	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCBR0002					204 ms	NONE	0.00	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVP60_3					203 ms	NONE	0.44	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVPFRK1					202 ms	NONE	0.00	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVPFRK2					203 ms	NONE	0.00	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVPFRK3					203 ms	NONE	0.00	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVWSL_3					203 ms	NONE	4.47	NONE	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVWSL_4					203 ms	NONE	0.00	S4	NONE	01.01.70	01:00:00	SERVER					
<input type="checkbox"/>		TCVWSL_5					202 ms	NONE	0.00	N6	NONE	01.01.70	01:00:00	SERVER					

Figure 3: Ressource table Targets

The column  indicates whether there is a connection with the target. A green icon indicates regular contact between the target and TransportControl. The exact time interval of the received UDP signals can be read out in the Response column. The light in the column  switches to red when the maxOfflinePeriod defined in the model is exceeded. A yellow light means that the received telegram cannot be processed. Reasons for this could be an incorrect or faulty input script or model. The lights in the column  turn red when there is an alarm on the corresponding target. Which status and error messages lead to an alarm (alert) is defined in the model. By clicking on the target, the exact status and error messages of the target can be found. In addition, the network address with IP address and port as well as the first login of the target is also displayed. The column  provides information on if targets were stopped manually. This is indicated by a red light. Further information can be found in Chapter 3.4 "Controlling targets". A yellow light in the column  shows if the target is currently being remotely controlled. Simulated targets can be distinguished from real targets by the column . A blue light represents a simulated target, a gray light represents a real target.

The **Update** column indicates if the target is currently receiving updates via a special update telegram. The transmission of an update can be performed via the **Update** button. **Mileage [km]** indicates the distance covered in kilometers since the target has been online. The **Job time** serves as a timer and counts up only if an event of the type Work is active. The **Job time** is only reset when the Work type event ends. This makes it possible to determine how long a vehicle has already been assigned to process a TO or how long the vehicle has not processed a TO. This value can then be used in scripts to implement certain time-dependent actions. The **Location** shows where the vehicle is currently located. The **Destination** indicates the destination to which the target is currently traveling. The **Modified** column indicates when the target was last edited. The **Responsibility** shows UDP as responsible until a user processes the target.

3.4. Control targets

To stop a single target or multiple selected targets, the  button is pressed. When the button is activated, all selected vehicles stop, and no further action commands will be sent. A stopped target is displayed flashing red on the map. If the  button is pressed again, the vehicle resumes operation.

In addition, it is possible to remotely control simulated and real targets via the **Remote** button. If remote control of the target is possible, a window opens with the control and a yellow flashing circle appears around the vehicle.

The requirement for this is that the vehicle type of the target supports this. This property is set model-specifically in the Properties of the vehicle models. If the vehicle type does not allow remote control, the message “Access denied” appears.

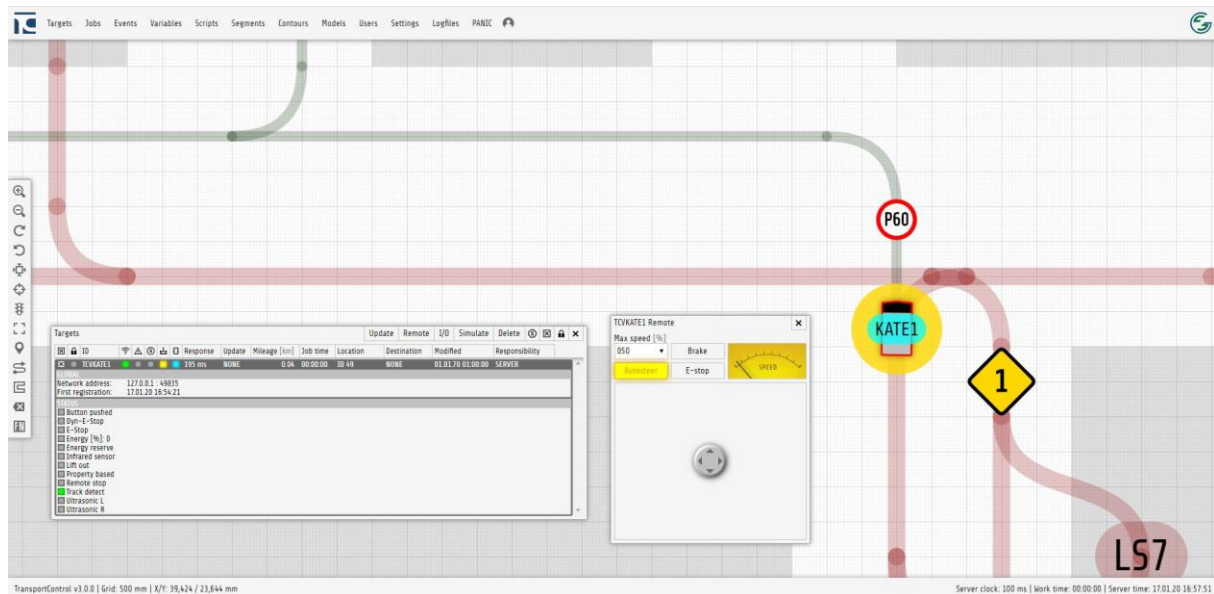


Figure 4: Remote control

In the remote window, the maximum speed in percent can be selected via drop-down menu. There are also three buttons and a joystick available. The joystick is used to determine the speed and direction of the vehicle. To do this, the joystick is moved in the desired direction. The three buttons Brake, E-Stop and Autosteer only have a function if they are physically supported by the selected target and can be executed. The brake can be applied by using **Brake**. **E-Stop** (Emergency-Stop) is the emergency stop of the target. If the **Autosteer** button has been activated, the vehicle automatically follows the segments. By moving the joystick to the left, right or straight ahead, you can decide in which direction the target turns at crossroads. Only one target can be remote controlled per user at a time.

3.5. Traffic management

Traffic control within TransportControl is realized by envelopes. Length, width and resolution of the envelope curves can be defined individually for each model (see chapter 10.2). The envelopes are used to predict the future position of each target in order to avoid collisions through targeted braking before the vehicle sensors are triggered. The envelope curve of each target changes depending on the speed and the track.

If the envelope of a target is violated, the target brakes. A violation occurs either through contact with the envelope curve of another target or with a stop sign. If the violation of the envelope is not removed, the target will decelerate to a stop. If the violation of the envelope is removed during deceleration, the target accelerates again.

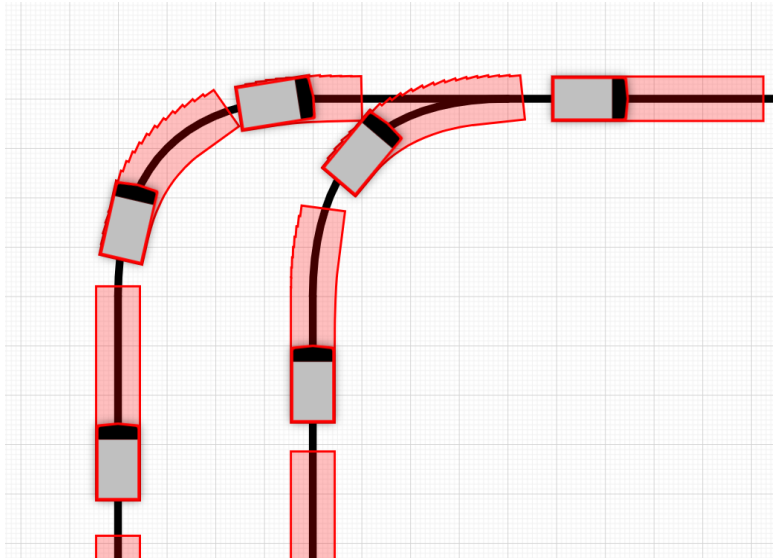


Figure 5: Dynamic traffic control through envelope curves

The envelope not only provides physical protection of the vehicle and traffic control for all targets in TransportControl, but also serves to protect any payloads that extend beyond the dimensions of the target. The dynamic traffic flow that results from the use of the envelope also allows for maximum route usage.

ATTENTION: The envelope curves depend on the accuracy of the transmitted coordinates of the target. The envelopes can therefore not replace the safety sensors of the vehicles!

4. Segments

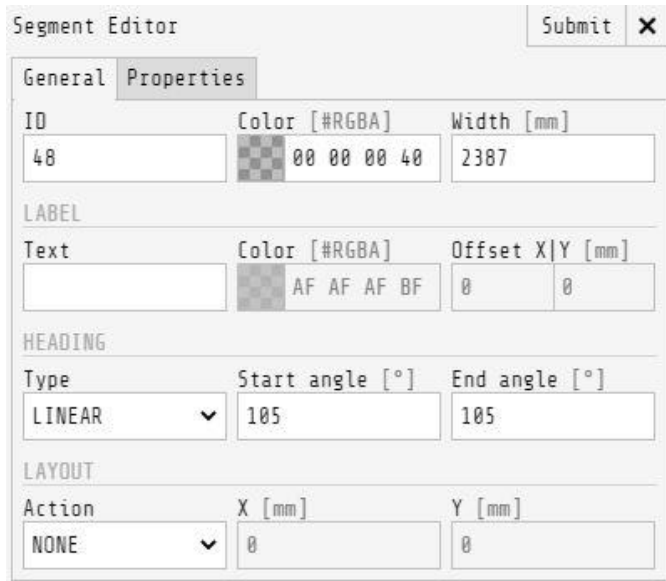
A segment represents a virtual route section. Each segment has certain properties that affect either the behavior of the targets or the visualization in the map.

Segments												Search	Merge	Split							
		ID	Label	Parts	Length [m]	V absolute	V relative	Condition				Modified	Responsibility								
		39	NONE	5	57.45	auto	100 %	100 %				11.01.21 11:51:01	ADMIN								
		40	NONE	5	57.45	auto	100 %	100 %				11.01.21 11:51:32	ADMIN								
		42	NONE	5	55.20	auto	100 %	100 %				11.01.21 11:51:32	ADMIN								
		43	NONE	5	42.65	auto	100 %	1 %				11.01.21 11:52:46	ADMIN								
		44	NONE	8	54.46	auto	15 %	100 %				11.01.21 10:29:31	ADMIN								
		45	NONE	10	89.15	auto	100 %	100 %				11.01.21 11:51:32	ADMIN								
		46	NONE	6	105.31	89 m/s	100 %	100 %				11.01.21 11:50:49	ADMIN								
		47	NONE	8	103.23	auto	100 %	100 %				11.01.21 11:51:32	ADMIN								
		48	NONE	9	80.34	auto	100 %	100 %				11.01.21 11:51:32	ADMIN								
		49	NONE	10	99.54	auto	67 %	100 %				11.01.21 10:29:43	ADMIN								
		50	NONE	6	43.24	auto	100 %	100 %				11.01.21 11:52:16	ADMIN								
		51	NONE	6	44.25	auto	100 %	100 %				11.01.21 10:30:10	ADMIN								
		52	NONE	8	69.91	auto	100 %	100 %				03.11.20 13:31:41	New_User								
		53	NONE	1	243.60	auto	100 %	40 %				11.01.21 11:52:38	ADMIN								
		54	NONE	8	82.31	auto	100 %	100 %				03.11.20 13:31:41	New_User								
		55	NONE	1	141.38	auto	100 %	100 %				11.01.21 10:29:57	ADMIN								
		56	NONE	1	128.71	auto	100 %	100 %				11.01.21 11:52:16	ADMIN								
		57	Lager	1	111.90	auto	100 %	100 %				11.01.21 11:53:56	ADMIN								
		59	LS1	1	117.68	auto	100 %	100 %				11.01.21 11:53:44	ADMIN								
		60	LS2	7	76.13	auto	100 %	100 %				11.01.21 11:54:05	ADMIN								
		61	NONE	1	104.82	auto	100 %	100 %				11.01.21 11:52:03	ADMIN								
		62	NONE	1	162.22	auto	100 %	100 %				11.01.21 11:52:03	ADMIN								
		63	NONE	1	444.77	auto	100 %	2 %				11.01.21 11:52:55	ADMIN								
		64	NONE	1	379.36	auto	100 %	100 %				11.01.21 11:52:03	ADMIN								
		65	NONE	1	87.43	auto	100 %	100 %				11.01.21 11:52:03	ADMIN								
		66	NONE	1	257.14	auto	100 %	100 %				11.01.21 11:52:03	ADMIN								

Figure 6: Resource table Segments

Each segment is assigned a unique ID. IDs of existing segments cannot be changed by editing. If a segment with a specific ID is to be created, it can be created and the ID entered if it does not already exist.





The screenshot shows the 'Segment Editor' window with the 'General' tab selected. The window has a 'Submit' button and a close 'X' button. The 'General' tab contains several sections: 'ID' with a text field containing '48'; 'Color [RGBA]' with a color picker and a text field containing '00 00 00 40'; 'Width [mm]' with a text field containing '2387'; 'LABEL' section with 'Text' (empty), 'Color [RGBA]' (AF AF AF BF), and 'Offset X|Y [mm]' (0 | 0); 'HEADING' section with 'Type' (a dropdown menu showing 'LINEAR'), 'Start angle [°]' (105), and 'End angle [°]' (105); and 'LAYOUT' section with 'Action' (a dropdown menu showing 'NONE'), 'X [mm]' (0), and 'Y [mm]' (0).

Figure 7: Segments Editor – General

In the Segment Editor, all properties of a segment or multiple segments can be displayed and modified. The Segment Editor is divided into the **General** and **Properties** tabs.

In addition to the subsequently unchangeable ID, the color and width of the segment can be adjusted. Below that, you can define the properties of the segment label. If no entry is made in the **Text** field, no label will be displayed on the map and the resource table will show the entry **NONE**.

Under **Heading** the necessary alignment of the target on this segment is defined. In the Type drop-down menu, the selection options **Forward**, **Backward**, **Fixed**, **Linear** and **question mark "?"** are available. In addition, there are two input fields for the start and end angle in degrees. With the type **Forward** a forward directed drive of the targets is determined for this segment. With **Backward** a backward directed drive. The **Fixed** type leaves a vehicle aligned in one direction during the entire journey, regardless of the segment course. The angle is entered in the **Start angle [°]** field. The entry for the end angle is locked since this is equivalent to the start angle because of the type. By using the **Linear** type, the target changes its orientation linearly over the complete length of the segment. Start and end angles must be entered for this. The angle specification always refers to a target directed in positive X axis. For the use of each type, it must be ensured that the vehicle can also technically perform the corresponding action.

When editing several segments of the same type, this is displayed accordingly. If several segments of different types are being processed, this is indicated by a question mark "?". If a selection is now made in the dropdown, all selected segments will be changed. In the Layout area, different adjustments of the segment can be implemented. Segments can be inverted, moved or mirrored on the basis of one of the axes. To do this, one of the commands from the dropdown list must be selected and, if necessary, the X and Y coordinates entered in mm.

The **Properties** tab lists all properties that have an effect on the vehicle behavior. If internally used properties are deleted, they will be added again with the next editing and the default value will be entered.









Segment Editor Submit X

General Properties

NAME	VALUE		
condition	100	+	-
left	false	+	-
spotTurn	false	+	-
closure	false	+	-
relativeSpeed	100	+	-
stop	-1	+	-
absoluteSpeed	auto	+	-
right	false	+	-
prohibition	NONE	+	-
permission	TCV	+	-

Figure 8: Segment Editor - Properties

The following table visualizes the graphical representation of the different properties of a segment.

Name	Map visualization	„Value“ Entries	Explanation
closure		false, true	Closes the segment, so that this segment is no longer taken into account in route finding. If there are already targets on this segment or if no alternative route is available to complete the TA, the target stops. Error message in the job manager for the affected TO: "Can't find route for current order".
condition		0 – 100	Defines the state of the segment in percent. The initial value is 100, which is why no sign is displayed at this value. As lower the value is, as more worse the condition is. Can be used to give preference to certain routes over other routes.
relativeSpeed		0 – 100	Percentage speed limit related to the maximum speed of each vehicle type.
spotTurn		false, true	Allows targets to turn in place at the beginning of the segment - Requirement: the vehicle must be technically able to process and physically execute this signal accordingly.
prohibition		1 Target Key	Prohibits the use of this segment for targets for which the target key starts with the entered value.
permission		1 Target Key	Allows the use of this segment with targets where the target key starts with the entered value.



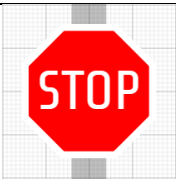
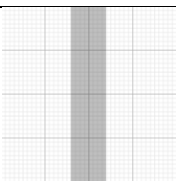







stop		-1, 0 – 100	Makes a stop sign appear at a specified position of the segment. The position of the sign depends on the specified progress in percent (0-100). At -1, no sign is generated. The segment is still included in the route finding. All vehicles stop directly in front of the sign.
absoluteSpeed		auto, Numerical value	Speed limitation of all vehicle types to a single speed in m/s. This limit is not displayed on the map.


Table 1: Segment Properties, presentation and effects

Except for the **permission** and **prohibition** properties, all properties are also displayed in the **Segments** resource table. For this purpose, in addition to the corresponding text, the symbols , ,  and for the properties **spotTurn**, **closure** and **stop** will be used.

To draw segments, press the  button. By holding down the left mouse button, you can now draw segments directly in the map. Via , the last drawn segment can be removed. It is also possible to force a straight connection between the start and end points of a segment by pressing  or the "Shift" key during drawing. While drawing, the current length of the segment is displayed in the footer. To exit the drawing mode, it is necessary to press  again. After that, the **Segment Editor** opens, where the properties of all just created segments are defined. Confirming the entries by pressing **Submit** completes the drawing.

The **Merge** button can be used to merge several connected segments into a single segment. All settings (ID, color, width, label, heading and properties) of the last segment selected in the row are adopted.

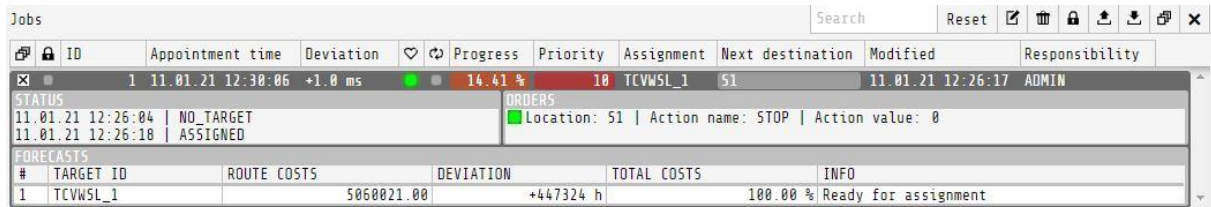
The **Parts** column in the resource table shows how many segments have been combined to form a segment. It is also possible to split the segment again by clicking the **Split** button. The length, shape and curve of the previously joined segments are then recovered, but the properties are the same as those of the joined segment, because TransportControl does not save the previously rejected data.

If the **Clone** button is pressed, the Segment Editor opens. Changes can be made here. After confirming the settings in the Segment Editor with **Submit**, a copy of the previously selected segments is created, which only differs by its ID. Über die Schaltfläche  lassen sich Segmente ohne Aufruf des Segment Editors öffnen und schließen.

5.Jobs

All transport orders are managed via the **Jobs** resource table. If a job is selected, the individual orders and the status of the job are displayed. Orders are the individual subtasks of a transport order.

The job manager not only allows transport orders to be processed directly, but also allows jobs that are due in the future to be managed. The time at which the target should be at a specific destination can be precisely defined by setting the date and time.



ID	Appointment time	Deviation	Progress	Priority	Assignment	Next destination	Modified	Responsibility
1	11.01.21 12:30:06	+1.0 ms	14.41 %	10	TCVWSL_1	S1	11.01.21 12:26:17	ADMIN


STATUS	
11.01.21 12:26:04	NO_TARGET
11.01.21 12:26:18	ASSIGNED

ORDERS	
Location: S1	Action name: STOP Action value: 0


FORECASTS					
#	TARGET ID	ROUTE COSTS	DEVIATION	TOTAL COSTS	INFO
1	TCVWSL_1	5060021.00	+447324 h	100.00 %	Ready for assignment

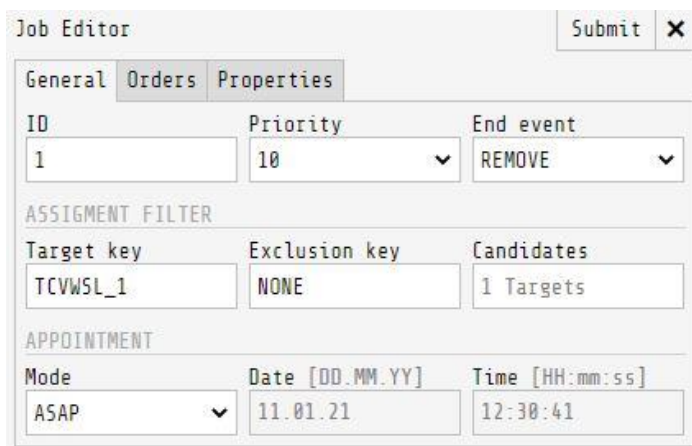
Figure 9: Ressource table jobs

The resource table Jobs shows the IDs, the calculated arrival time and the current deviation from the arrival time.

The  icon indicates the status of the job. A green light means that the job is running and will be completed in time. A yellow light indicates that the job will be completed later than planned. However, the light only turns yellow when a certain deviation is exceeded, depending on the total runtime of the job. A gray light means that the job has not yet started or has already been completed. A red light means that the travel job cannot currently be executed.

Furthermore, the **Forecast** of the selected job is displayed. This is the pre-calculation of the job, which takes into account the routes costs and total costs. It also shows a possible delay and the availability of the AGV for this job. These parameters are calculated and taken into account for all vehicles and then the appropriate, most suitable, vehicle is assigned.

The  column shows whether a job is repeated after it has been processed. Blue stands for the repetition (**End event** = Restart), gray for the removal of the job after processing (**End event** = Remove). The **Progress** shows the percentage progress of the transport request. This is also displayed for each target on the map when the resource table is open. The priority defines the urgency of the TO and influences the processing sequence. Furthermore, the **target key**, the executing vehicle, the time of the last processing and the responsibility are displayed. In addition, the assigned vehicle, the next segment to be approached, a time stamp of the last processing and the responsible user can be seen.



Job Editor Submit X

General | Orders | Properties

ID	Priority	End event
1	10	REMOVE



ASSIGNMENT FILTER

Target key	Exclusion key	Candidates
TCVWSL_1	NONE	1 Targets

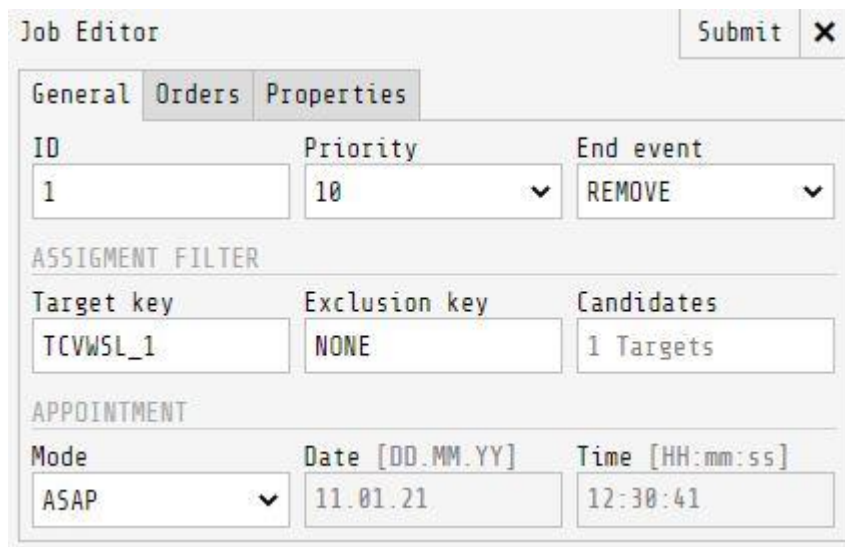
APPOINTMENT

Mode	Date [DD.MM.YY]	Time [HH:mm:ss]
ASAP	11.01.21	12:30:41

Figure 10: Job Editor - General

The **Job Editor** opens as soon as a new job is created via the  button. In addition, it is also displayed when an existing job is selected and edited via the  button. The Job Editor is divided into the three tabs **General**, **Orders** and **Properties** and allows to configure the job.




The priority is selected via a drop-down list. The entry in the Target Key field defines the targets by which this transport request can be executed. Individual vehicles or vehicle groups can be excluded via the **Exclusion key**. The **Candidates** field shows how many vehicles will be considered for job allocation. In the drop-down list for the **End Event**, **RESTART**, **REMOVE** or **ARCHIVE** are available for selection. **RESTART** processes the individual orders of the job in a loop. **REMOVE** deletes the job after all orders have been processed. The **ARCHIVE** selection leaves the job in the Jobs resource table after it has been processed. It can then be restarted via the **Reset** button. The time mode of the transport job can be set under **APPOINTMENT**. **ASAP** stands for immediate processing. **FIXED** enables an entry to be made in the Date and Time fields. The time entry refers to the arrival time.



The screenshot shows the 'Job Editor' window with the 'Orders' tab selected. The window has a title bar with 'Job Editor' and a 'Submit' button with a close icon. Below the title bar are three tabs: 'General', 'Orders', and 'Properties'. The 'Orders' tab contains several sections:

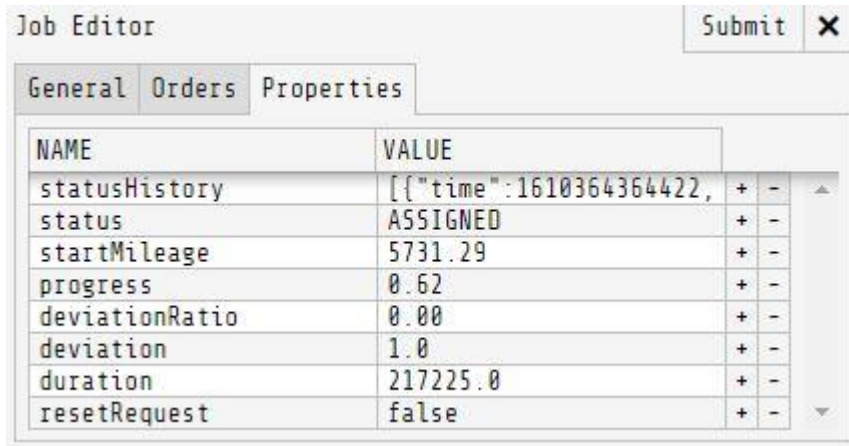
- General Fields:** ID (1), Priority (10), and End event (REMOVE).
- ASSIGNMENT FILTER:** Target key (TCVWSL_1), Exclusion key (NONE), and Candidates (1 Targets).
- APPOINTMENT:** Mode (ASAP), Date (11.01.21), and Time (12:30:41).

Figure 11: Job Editor - Orders

All individual steps of the TO are displayed under Orders. Each line in the table represents a partial order. Via  a new line can be added and via  the corresponding line is removed. For each **Order**, the target segment must be specified, the **Action** upon arrival at the target segment must be defined, and the **ACTION VALUE** must be set. This means that even complex processes can be implemented within a job. By selecting an order in the column , you can specify the single step for which the **Date** and **Time** information from the **General** tab is to be used. This selection has an impact on the progress indicator, since it refers to the order to be fulfilled and not to the entire TA. The **Deviation**, which is specified in the resource table, also refers to this specification. If it is fulfilled, no further delay is added.

Via the dropdown list in the **ACTION NAME** column, the options **NONE** and **STOP** can be selected by default, but any other action defined in the model is also displayed. In order for the actions of a model to be displayed in the drop-down list under **ACTION NAME**, it is necessary that the corresponding target key has been entered in the **General** tab. If you leave the input field under **ACTION VALUE** empty, TransportControl shows which entries are possible. These refer to the entry of the minimum and maximum value of each individual action. The entry **NONE** refers to orders that are fulfilled by reaching the target segment. The **STOP** action causes a target to stop at the end of the selected segment.





NAME	VALUE			
statusHistory	[{"time":1610364364422,	+	-	▲
status	ASSIGNED	+	-	
startMileage	5731.29	+	-	
progress	0.62	+	-	
deviationRatio	0.00	+	-	
deviation	1.0	+	-	
duration	217225.0	+	-	
resetRequest	false	+	-	▼

Figure 12: Job Editor - Properties

The properties of the transport order are defined under **Properties**. These are filled automatically after the creation of the TO and adjusted accordingly during the runtime of TransportControl. Via **resetRequest** the job can be set to reset by script. This can be equated with the function of the **Reset** button from the Jobs resource table. The entry status shows the **Status** of the TO. Possible statuses are **NO TARGET**, **NO ROUTE**, **SCHEDULED**, **ASSIGNED**, **DELAYED** or **COMPLETED**. The properties **startMileage**, **progress**, **deviationRatio**, **deviation** and **duration** are job information that can be accessed via script.

TransportControl tries to process all jobs optimally, depending on the costs of the transport orders. The goal is to execute all jobs with the lowest possible costs. The costs of a job consist of the information about the route (distance, speed, route condition) as well as the arrival calculated at the time of creation. The priority has the effect that with increasing priority, the costs for a deviation from the calculated arrival increase and the costs for the route decrease. If the priority decreases, the opposite is the case.


If several targets are available for a job, TransportControl selects the target depending on the calculated costs. If fewer targets than TO are available, TransportControl assigns the job with the closest **Appointment Time** in the future first. The **Appointment Time** of jobs to be executed immediately is the time of creation or the last editing. For job assignment, not only those targets are taken into account who currently do not have a job, but also targets who will be unemployed in the future.

If the **Appointment Time** is the same for two or more jobs, the job with the highest priority is preferred. If the priority is also identical, the TO that was created first is processed first. The creation of a transfer order is tracked to one hundredth of a second. TransportControl does not assign **Creation Time** twice. In addition, the stagnation of job progress flows negatively into the calculations of subsequent jobs and therefore influences the assignment of the TO if a target is blocked.



6. Contours

Contours are used to represent the operating environment of the AGV in two dimensions and to create a way of orientation for the user. Polygonal surfaces are used for this purpose. It is not possible to represent curves or circles using contours.

For the creation of contours  must be pressed in the toolbar. With each mouse click, a black point is placed on the grid of the map. These points are always connected to the previous point.

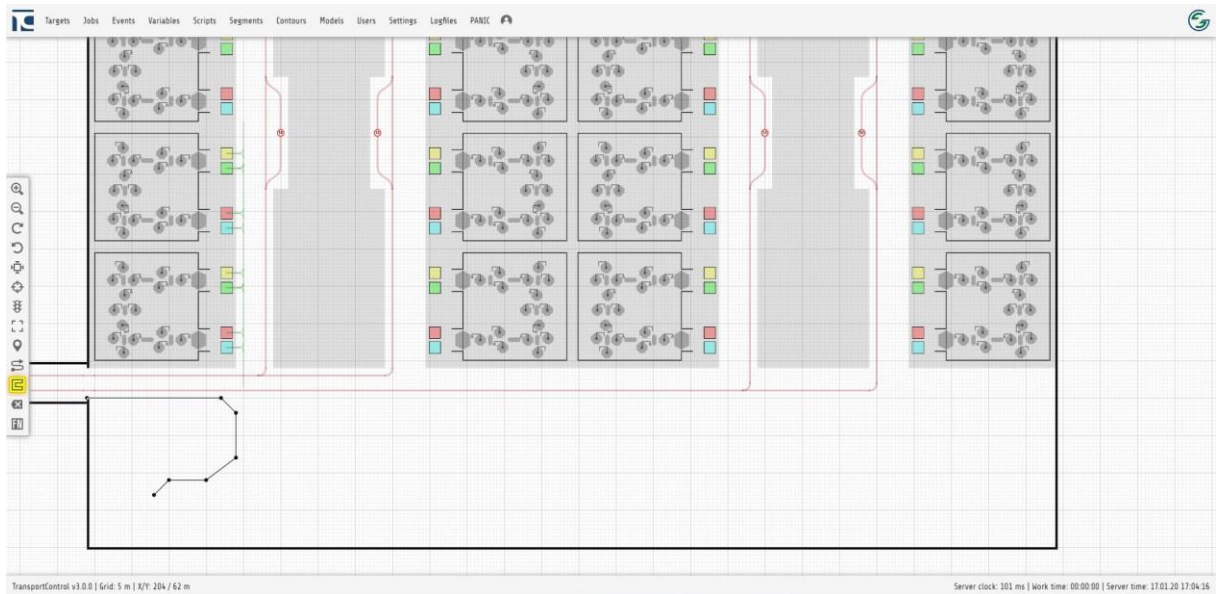




Figure 13: Set contours points

If the last contour point is connected to the start point, the shape closes and further contours can be drawn or drawing can be terminated. If an error occurs during drawing, the individual steps can be undone via . If a contour has already been connected to its starting point, the  button deletes the contour completely.

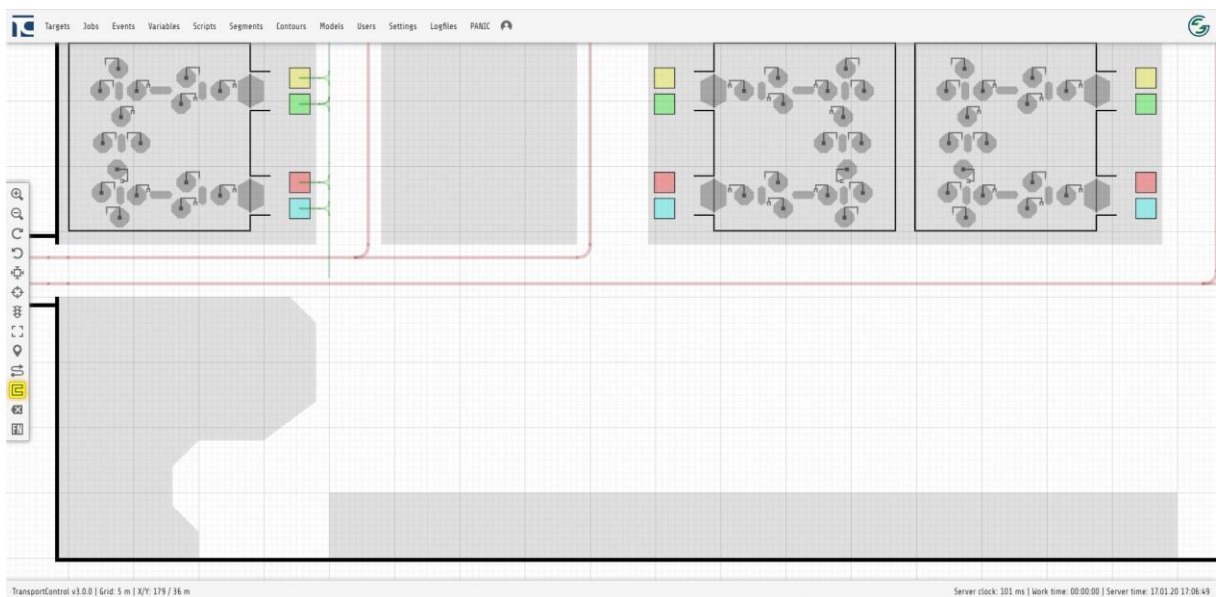




Figure 14: Draw contour

When all the desired contours have been drawn, click again  in the toolbar. The **Contour Editor** then opens, where the properties of all previously drawn contours can be defined. The drawn contours are accepted with **Submit**.

When all the desired contours have been drawn, click again  in the toolbar. The **Contour Editor** then opens, where the properties of all previously drawn contours can be defined. The drawn contours are accepted with **Submit**.

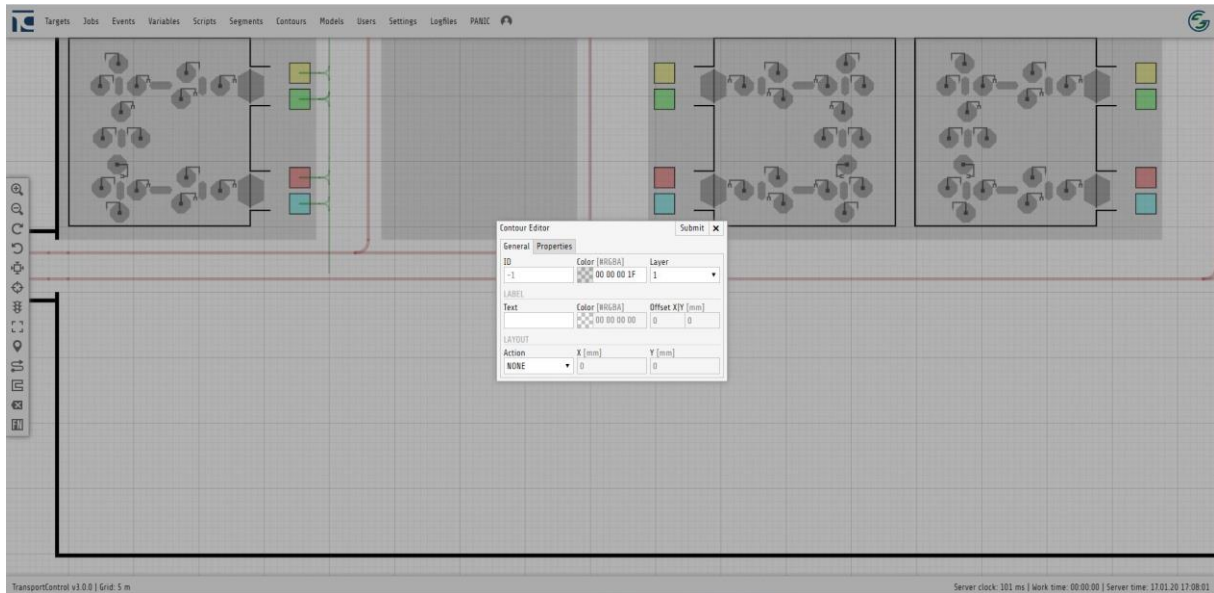


Figure 15: Contour Editor

In the General tab of the **Contour Editor**, properties such as contour color, label text and label color are defined, or adjustments such as shifting or mirroring along the axes are made. In addition, contours can also be assigned to a specific **Layer**. There are four different layers (0-3), which can be switched on and off via the resource table. In the Properties tab, additional entries can be made, which can be read, processed or overwritten via script, interfaces or the REST API.

Contours

Search

0123

		ID	Label	Layer	Points	Area [m²]	Color	Modified	Responsibility
<input type="checkbox"/>		0	NONE	0	7	11173.34	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		1	NONE	0	5	3615.61	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		2	NONE	0	5	20611.01	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		3	NONE	0	5	36040.25	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		4	NONE	0	5	14894.35	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		5	NONE	0	6	30444.05	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		6	NONE	0	5	9552.00	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		7	NONE	0	5	9125.57	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		8	NONE	0	5	16331.88	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		9	NONE	0	7	16497.32	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		10	NONE	0	5	795.65	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		11	NONE	0	5	21420.34	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		12	NONE	0	5	45832.10	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		13	NONE	0	5	17487.99	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		14	NONE	0	5	54310.09	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		15	NONE	0	5	16013.12	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		16	NONE	0	5	15484.77	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		17	NONE	0	5	15773.94	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		18	NONE	0	5	12513.27	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		19	NONE	0	5	12269.99	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		20	NONE	0	5	13965.21	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		21	NONE	0	5	14683.25	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		22	NONE	0	6	104177.	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		23	NONE	0	5	12577.98	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		24	NONE	0	7	44325.57	00 00 00 40	11.01.21 16:37:59	ADMIN
<input type="checkbox"/>		25	NONE	0	5	20105.00	00 00 00 40	11.01.21 16:37:59	ADMIN

Figure 16: Resource table Contours

All contours are displayed in the **Contours** resource table. In addition to the ID, information such as the label, the layer, the set color and the responsibility are displayed there. It also shows the number of points each contour consists of and the area it occupies. The **Edit**, **Remove** and **Clone** buttons provide the functions already described. Via the numbers 0, 1, 2 and 3 in the upper right area, each layer can be switched on and off individually. Layers that are switched on have a dark gray background. Deactivated layers are displayed in light gray.




Via the **Upload** field in the **Contours** resource table, it is possible to integrate dxf files or previously downloaded contours directly into TransportControl. Integrating dxf files usually requires prior editing of the file, since many layouts have too many details and the total size of the file should not significantly exceed 1 MB. TransportControl analyzes the transmitted data and converts them into contours. Connected areas in the dxf file are recognized better than individual lines. It is generally not possible to transfer fonts and round shapes.

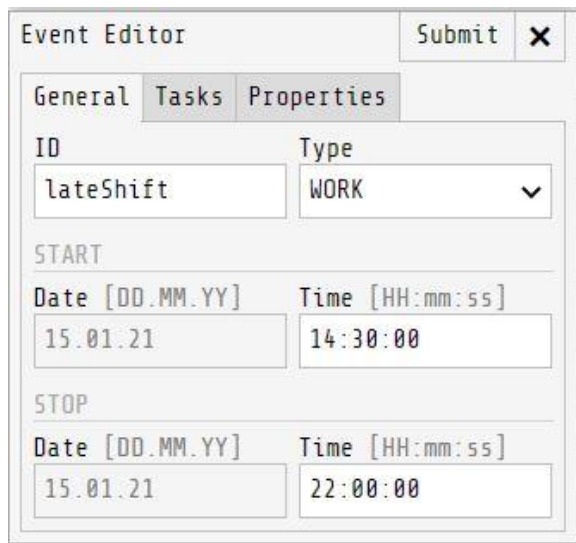
7.Events

Recurring and one-time events can be managed via the event system. By using events time-controlled actions can be implemented or, on their basis, time-controlled actions of scripts can be initiated.

Search																
		ID	Type	Start	Stop		Modified	Responsibility								
<input type="checkbox"/>		early_shift	WORK	15.01.21 06:00:00	15.01.21 14:30:00		15.01.21 10:27:37	ADMIN								
<input type="checkbox"/>		earlyShiftBre...	BREAK	15.01.21 10:30:00	15.01.21 11:00:00		15.01.21 10:28:35	ADMIN								
<input type="checkbox"/>		lateShift	WORK	15.01.21 14:30:00	15.01.21 22:00:00		15.01.21 10:29:32	ADMIN								
<input type="checkbox"/>		lateShiftBreak	BREAK	15.01.21 18:30:00	15.01.21 19:00:00		15.01.21 10:30:18	ADMIN								
<input type="checkbox"/>		construction5...	SINGLE	12.02.21 06:00:00	12.02.21 22:00:00		15.01.21 10:31:11	ADMIN								

Figure 17: Resource table events

The name, type, start and end time, activity status and the last processing time can be taken from the resource table. The column  shows via a light which events are currently active. Use  to create a copy of all events listed in the table. Via  a file created in this way can be uploaded at a later time or in another system.



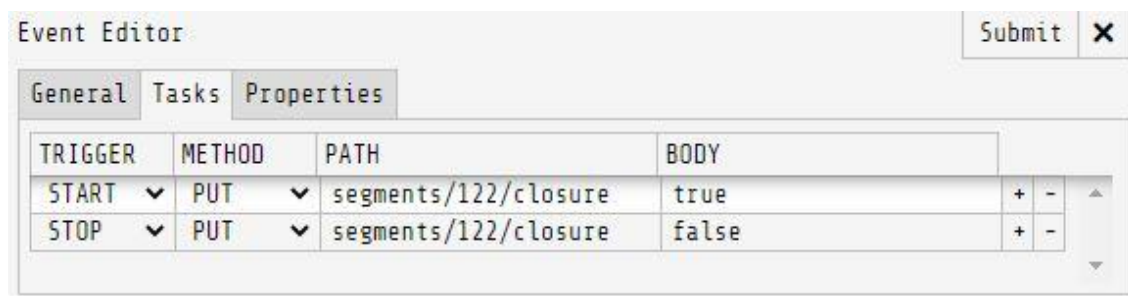
The 'General' tab of the Event Editor shows the following fields:

- ID:** lateShift
- Type:** WORK (dropdown menu)
- START:**
 - Date [DD.MM.YY]:** 15.01.21
 - Time [HH:mm:ss]:** 14:30:00
- STOP:**
 - Date [DD.MM.YY]:** 15.01.21
 - Time [HH:mm:ss]:** 22:00:00

Figure 18: Event editor – General

Events are created or edited in the **Event Editor**. Events are divided into three types: **WORK**, **BREAK** and **SINGLE**.

The types **WORK** and **BREAK** refer to daily repeating events. For this reason, a date specification is blocked when these types are selected. **WORK** and **BREAK** are used for the inclusion of a shift system or a clocked production sequence. For **WORK** events the **WORK TIME** runs in the specified time period. In order to consider operational interruptions (e.g. breaks, maintenance), the type **BREAK** is used. In the period specified in a **BREAK** type event, the work time does not continue to count. The **Work time** is displayed in the info bar on the right side and can be used within scripts to time different tasks. The **SINGLE** type is used for one-time events. For this purpose, the date and time of the start and end date are specified. After reaching the end date the event will be deleted.



The 'Tasks' tab of the Event Editor shows a table with the following data:

TRIGGER	METHOD	PATH	BODY			
START	PUT	segments/122/closure	true	+	-	▲
STOP	PUT	segments/122/closure	false	+	-	▼

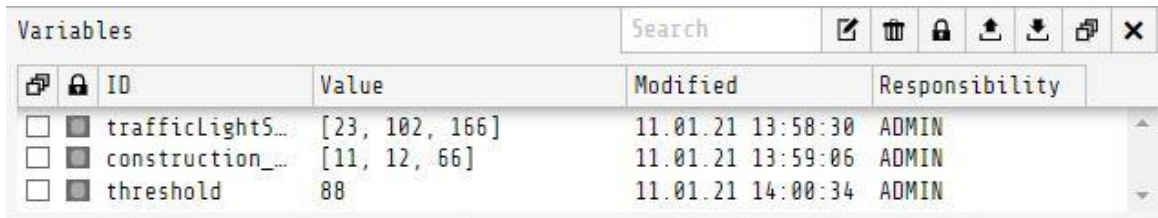
Figure 19: Event Editor - Tasks

In the Tasks tab, all **REST API** requests can be linked to the **START** or **STOP** trigger. The trigger selection triggers the corresponding action either when the start deadline or the end deadline is reached. Details about the commands and inputs can be found in the **REST API** documentation. In the **Properties** tab, you can also make additional entries that can be read, processed or overwritten by **Scripts**, via interfaces or via the REST API.





8. Variables


Variables are used to make adjustments to customer-specific programmed functions (scripts) without requiring in-depth knowledge of programming. A requirement for this, however, is knowledge of which customer-specific functions have been implemented and what exact influence the variables have on these functions.

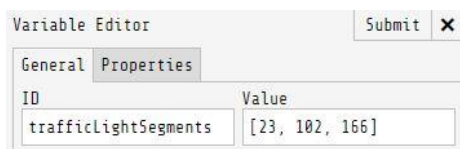


Variables		Search				
ID	Value	Modified	Responsibility			
trafficLightS...	[23, 102, 166]	11.01.21 13:58:30	ADMIN			
construction_...	[11, 12, 66]	11.01.21 13:59:06	ADMIN			
threshold	88	11.01.21 14:00:34	ADMIN			

Figure 20: Resource table variables

In the resource table **Variables**, the name and the value of the variable are displayed as well as the last editing time. After selecting an entry, the value is displayed in JSON format. Which **Value** or values must be entered in the variables depends on the scripts underlying it. A transfer of all created variables is possible via the  and  buttons.

Pressing the button  opens the variable editor. In the **General** tab, the name and the value or values of the variable are entered. A subsequent change of the name is not possible, because the name also represents the ID of the resource.



Variable Editor

Submit

General

Properties

ID

Value

trafficLightSegments

[23, 102, 166]

Figure 21: Variable Editor – General

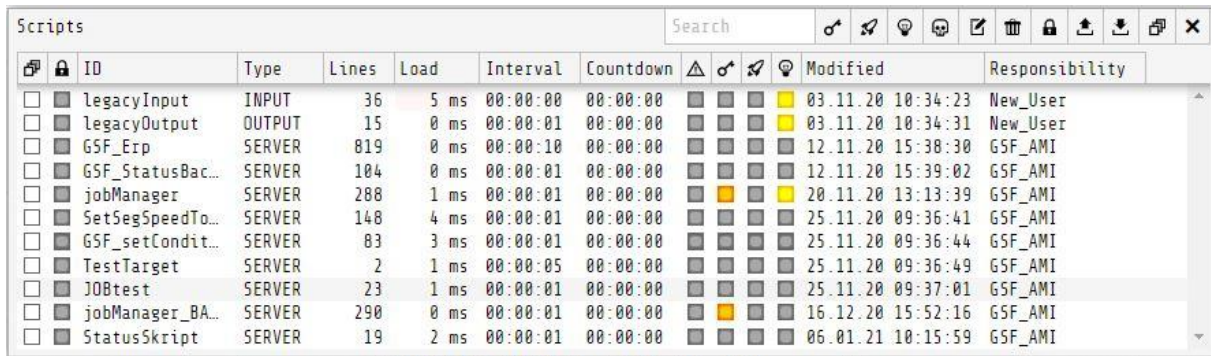
In the **Properties** tab, additional properties can be defined which can be read, processed or overwritten by scripts or via interfaces. Changes in the variable editor must be confirmed with **Submit**.

9. Scripts

A scripting system is integrated in TransportControl in order to be able to implement a customer-specific logistical and visual adaptation if required. The scripting system is based on the JavaScript programming language. It is divided into five different script types that allow different script areas to be customized. Changes are saved on the server and are always globally effective. Any errors during script execution are displayed in the resource table with the corresponding line number.



In addition to the standard functions such as adding, deactivating and deleting, the **Scripts** resource table also allows the upload and download of individual scripts.




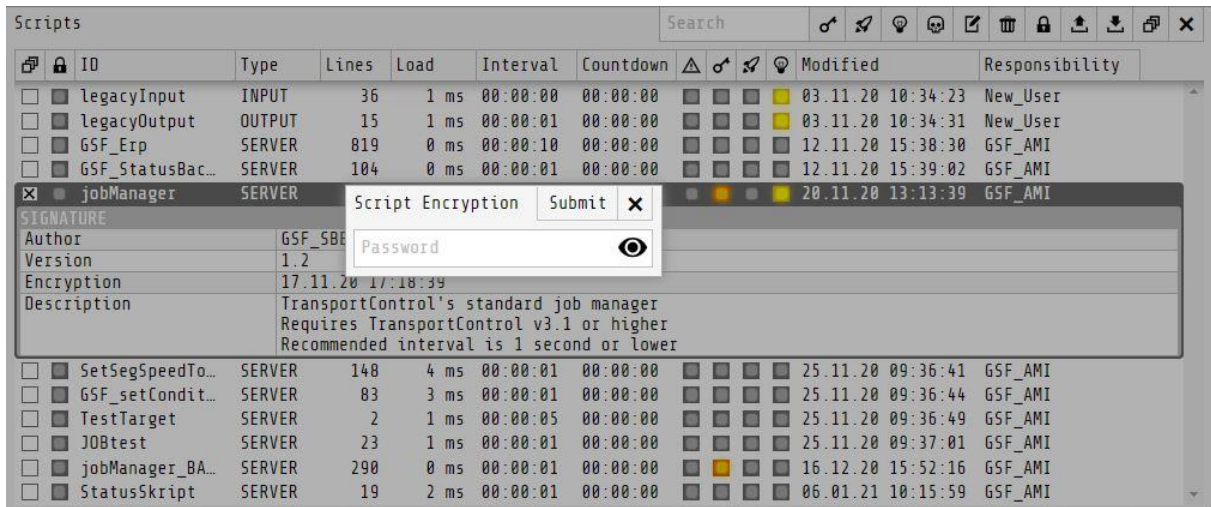


ID	Type	Lines	Load	Interval	Countdown	Modified	Responsibility
legacyInput	INPUT	36	5 ms	00:00:00	00:00:00	03.11.20 10:34:23	New_User
legacyOutput	OUTPUT	15	0 ms	00:00:01	00:00:00	03.11.20 10:34:31	New_User
GSF_Erp	SERVER	819	0 ms	00:00:10	00:00:00	12.11.20 15:38:30	GSF_AMI
GSF_StatusBac...	SERVER	104	0 ms	00:00:01	00:00:00	12.11.20 15:39:02	GSF_AMI
jobManager	SERVER	288	1 ms	00:00:01	00:00:00	20.11.20 13:13:39	GSF_AMI
SetSegSpeedTo...	SERVER	148	4 ms	00:00:01	00:00:00	25.11.20 09:36:41	GSF_AMI
GSF_setCondit...	SERVER	83	3 ms	00:00:01	00:00:00	25.11.20 09:36:44	GSF_AMI
TestTarget	SERVER	2	1 ms	00:00:05	00:00:00	25.11.20 09:36:49	GSF_AMI
JOBTtest	SERVER	23	1 ms	00:00:01	00:00:00	25.11.20 09:37:01	GSF_AMI
jobManager_BA...	SERVER	290	0 ms	00:00:01	00:00:00	16.12.20 15:52:16	GSF_AMI
StatusSkript	SERVER	19	2 ms	00:00:01	00:00:00	06.01.21 10:15:59	GSF_AMI

Figure 22: Resource table scripts

The current status of each script is shown in the column . A yellow light symbolizes an active script. If a script is executed at server startup can be seen in the column . The corresponding buttons in the edit bar can be used to enable/disable scripts or automatic startup on server startup. The **Load** column shows how long the last execution of the script took. The interval defines in which interval a script is executed again and the **Countdown** shows the time until the next execution. If a script contains an error, this is indicated in the **Error** column by specifying the line number. In that case, after selecting the corresponding script, a detailed error message will be shown.

By clicking on , the scripts can be encrypted. A window is opened in which a password for the respective script can be entered. After encryption, the script can only be decrypted again by using the defined password.



ID	Type	Lines	Load	Interval	Countdown	Modified	Responsibility
legacyInput	INPUT	36	1 ms	00:00:00	00:00:00	03.11.20 10:34:23	New_User
legacyOutput	OUTPUT	15	1 ms	00:00:01	00:00:00	03.11.20 10:34:31	New_User
GSF_Erp	SERVER	819	0 ms	00:00:10	00:00:00	12.11.20 15:38:30	GSF_AMI
GSF_StatusBac...	SERVER	104	0 ms	00:00:01	00:00:00	12.11.20 15:39:02	GSF_AMI
jobManager	SERVER	288	1 ms	00:00:01	00:00:00	20.11.20 13:13:39	GSF_AMI
SetSegSpeedTo...	SERVER	148	4 ms	00:00:01	00:00:00	25.11.20 09:36:41	GSF_AMI
GSF_setCondit...	SERVER	83	3 ms	00:00:01	00:00:00	25.11.20 09:36:44	GSF_AMI
TestTarget	SERVER	2	1 ms	00:00:05	00:00:00	25.11.20 09:36:49	GSF_AMI
JOBTtest	SERVER	23	1 ms	00:00:01	00:00:00	25.11.20 09:37:01	GSF_AMI
jobManager_BA...	SERVER	290	0 ms	00:00:01	00:00:00	16.12.20 15:52:16	GSF_AMI
StatusSkript	SERVER	19	2 ms	00:00:01	00:00:00	06.01.21 10:15:59	GSF_AMI

Script Encryption

Signature: GSF_SBB

Author: 1.2


Version: 1.2

Encryption: 17.11.20 17:18:39

Description: TransportControl's standard job manager
Requires TransportControl v3.1 or higher
Recommended interval is 1 second or lower

Submit

Figure 23: Resource table scripts encryption

By using , all running threads that occurred during the script execution are canceled. This should only be done if a person is familiar with the subject material. Since this can result in core exceptions.

Creating new scripts or editing existing ones should always be done by qualified personal only, as even small changes can have an impact on individual functions as well as the whole system.

When creating or editing a script, the **Script Editor** is opened. This is divided into the **General** and **Code** tabs.

In the **General** section the name (**ID**), the script type and the execution interval are specified. There are also the fields for the name of the creator (**Author**), version number (**Version**) and a script description (**Description**).

Script Editor

Submit X

General

Code

ID

jobManager

Description

TransportControl's standard job manager
Requires TransportControl v3.1 or higher
Recommended interval is 1 second or lower

Type

SERVER

Interval [HH:mm:ss]

00:00:01

Author

GSF_SBE

Version

1.2

Figure 24: Resource table scripts – General

An interval can be set only for scripts of the **SERVER** or **UPDATE** type. The name of a script is unchangeable after creation.

The **Code** tab displays the content of the script. The size of the **Script Editor** adjusts accordingly up to a certain size. After that, scroll bars can be used to navigate within the editor. After that, scroll bars can be used to navigate within the editor.

Script Editor

Submit X

General

Code

```

1 buffer.position(20);
2
3 // Attention: There is no get(byte dst), only get(byte[] dst)! This means
4 // buffer.get(t.in.speed) would use the absolute get(int index) overload.
5
6 input.speed = buffer.get();
7 input.steering = buffer.get();
8 input.segId = buffer.getShort();
9 input.segProgress = buffer.getShort();
10 input.searchSegLength = buffer.get();
11 input.searchSegId = buffer.getShort();
12 input.actionCode = buffer.get();
13 input.actionValue = buffer.get();
14 input.posX = buffer.getInt();
15 input.posY = buffer.getInt();
16 input.posH = buffer.getShort();
17 input.seqDataIndex = buffer.get();
18 input.seqDataValue = buffer.get();
19
20 //input.segId = -1;
21 //input.searchSegLength = 2;
22 //input.searchSegId = 18;
23
24 self.log("speed = " + input.speed);
25 self.log("steering = " + input.steering);
26 self.log("segId = " + input.segId);
27 self.log("segProgress = " + input.segProgress);
28 self.log("searchSegLength = " + input.searchSegLength);
29 self.log("searchSegId = " + input.searchSegId);
30 self.log("actionCode = " + input.actionCode);
31 self.log("actionValue = " + input.actionValue);
32 self.log("posX = " + input.posX);
33 self.log("posY = " + input.posY);
34 self.log("posH = " + input.posH);
35 self.log("seqDataIndex = " + input.seqDataIndex);
36 self.log("seqDataValue = " + input.seqDataValue);

```

Figure 25: Script Editor – Code

9.1. Typ SERVER

SERVER type scripts are called and executed from the central TransportControl Server. The logistics of a controlled plant can be adjusted by this. Many input signals have no predefined meaning within TransportControl and can be linked to actions as needed with the help of SERVER scripts. For example, transport orders can be generated or segments can be blocked if vehicles reach defined positions or stations are sending signals.

9.2. Typ UPDATE

Scripts of the **UPDATE** type are executed repeatedly within the Browser UI, taking into account the set interval, and are mostly used to visually adjust the map. Theoretically, every parameter and every function of the source code of the main UI can be manipulated or called from here.

9.3. Typ Custom

CUSTOM type scripts are executed in the case that a custom or mobile-optimized Custom UI is required. While the header and footer of the Custom UI are predefined, the content can be designed using a Custom script. When making changes to **CUSTOM** type scripts, a browser refresh of the Custom UI must be performed to show the changes.


9.4. Typ INPUT und OUTPUT

The types **INPUT** and **OUTPUT** are used to define the telegram structure for the communication between targets and TransportControl. To be able to integrate different FTF types in TransportControl, it is possible to define a separate input and output script for each model. Through the freely designable structure of the scripts, it is also possible to achieve compatibility with all FTF manufacturers. However, the requirement for this is the disclosure of the FTF telegrams by the manufacturer.

10. Models

10.1. General

A model must be defined for each participant that communicates with TransportControl via the UDP interface. Participants can be, for example, AGVs, manual vehicles or PLCs for fire alarm systems, traffic lights or gates. Without a model, TransportControl receives the data, but cannot assign or process it.

The model assigns the participant its representation on the map, the executable actions, the status and error messages and other properties. The **Model Editor** is opened when  is pressed. The name of the model, the target key and the input and output scripts are defined under the **General** tab.



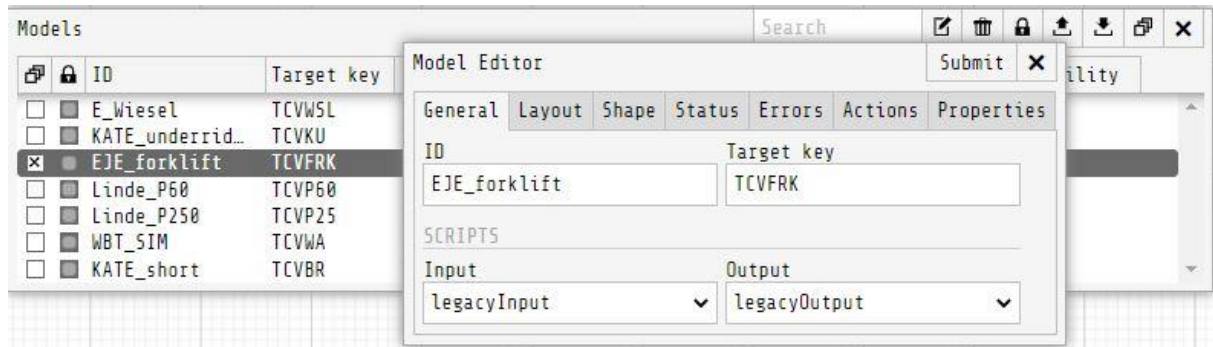


Figure 26: Model Editor – General

The name of a model is defined when it is created and cannot be changed afterwards. The function of the target key in relation to targets has already been explained in the chapter [Targets](#). The target key can be adjusted at any time. However, a change has the effect that the ID of all associated targets must be adjusted. The scripts used for the communication telegrams are selected under [Input](#) and [Output](#). In them is stored which information is transmitted at which position of the telegram. Since the vehicles send different information to TransportControl and TransportControl itself sends information back, a separate input and output script is required.

The [legacyInput](#) and [legacyOutput](#) scripts are used by default for simulated vehicles.

10.2. Layout

The Layout tab displays the target's layout and sets information for the animation on the map and for traffic control.

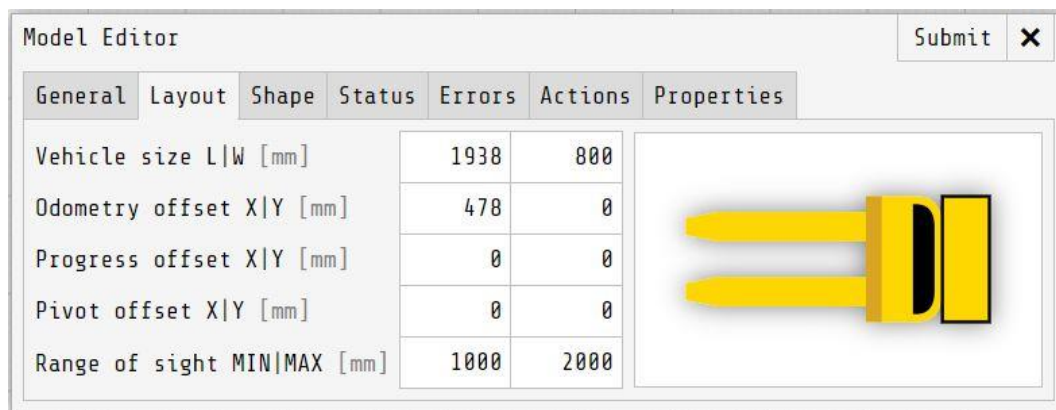


Figure 27: Model Editor – Layout

The [Vehicle Size](#) defines length and width of the vehicle. These values are used for the envelope calculation. If a target has a load that exceeds its length and width, a script can be used to adjust the [Vehicle Size](#) so that the excess load is taken into account in the envelope. The length of the envelope curve that looks ahead of the target is defined under [Range of Sight](#). The minimum envelope length is specified in the left text field, the maximum envelope length is specified in the right text field. The specification of the minimum envelope length refers to the visualization when the vehicle is at a standstill. The maximum envelope length corresponds to the length at maximum speed of the target. The [Odometry offset](#) defines to which point of the vehicle the transmitted coordinates of a real vehicle are related. The [Progress offset](#) defines the reference point of the segment progress relative to the center of the vehicle. This value is additionally definable, since the reference point is not always at the position of the localization unit due to internal processing in the vehicle system. The [Pivot Offset](#) describes the pivot point of the model on the map and therefore provides the correct optical representation by adjusting the movement of the model to the real vehicle.

10.3. Shape

The visual representation of the target on the map is defined in the **Shape** tab. Here you can create or edit the visual representation using the available tools. In order to ensure that the visualization of the vehicle is still sharp even at a large zoom level, a vector graphic is used for the visualization on the map. A code visualization is displayed in the lower area. All changes made to the visual representation affect the code and back again.

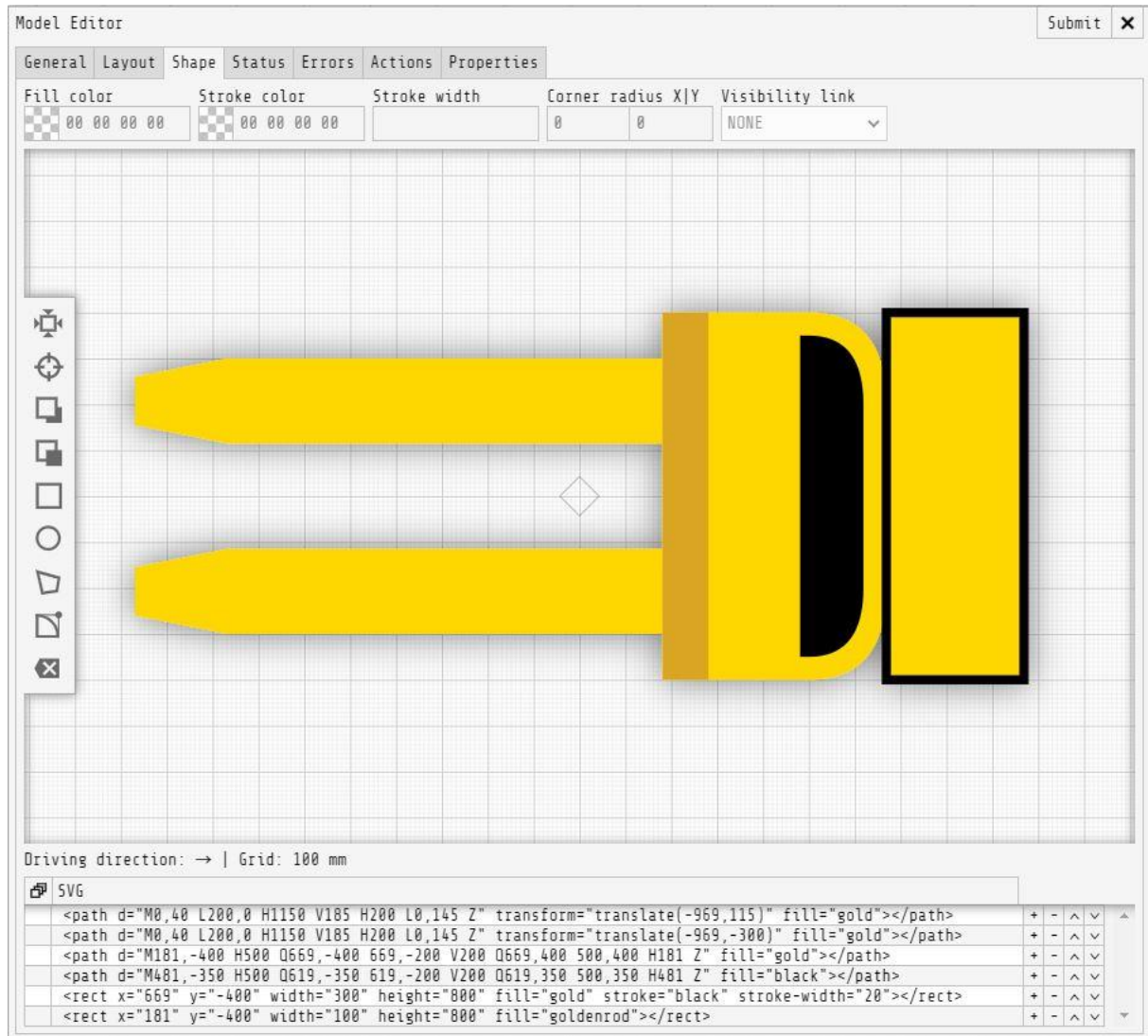




Figure 28: Model Editor – Shape


10.3.1. Tools




Reset view

Pressing  adjusts the zoom and position so that all elements are visible on the screen at the same time.


Zoom on selected elements

selecting one or more elements in the UI, clicking on  focuses the view on these objects. This affects both zoom and position of the image section.

forward

selecting an object, clicking on  the object moves back one level. This moves the line of SVG code down by 1 position.

backward

selecting an object, clicking on  moves the object one layer forward. This moves the line of code up by 1 position.

After

affects

Layer

After

code


Layer

After

SVG

Figure 29: Toolbar of the shape editor

Draw rectangle

After activating of  a rectangle can be created by mouse click and drag.


Draw circle

After activating of , a circle can be created by clicking and dragging.

Draw polygon

After activating , a polygon can be created by mouse clicks. This works analogously to drawing contours.

Draw curves

Activation of  allows to create round shapes. After setting three control points, the corresponding curve is automatically added. The first control point is the beginning of the curve. The third control point is the end. The second control point affects the gradient.

Undo last drawing action

Use  to undo the last drawing action or to remove an entire line of SVG code.

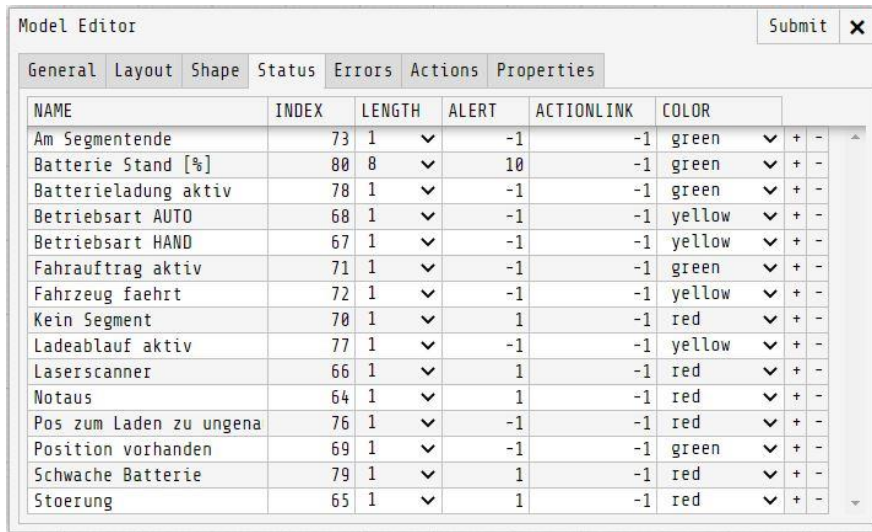
10.3.2. Adjustment of drawn shapes

After a drawn shape has been selected, the buttons for visualizing the shape can be used to make adjustments regarding shape and color. This includes the display color, the color of the shape border, as well as its width and the rounding of the corners. Via **Visibility link**, an entered status of the model can be used to display the currently selected shape or to hide it if the status is not set.

10.4. Status

In the **Status** tab, you can enter the statuses that are transmitted from the vehicle to TransportControl. **NAME**, **INDEX**, **LENGTH**, **ALERT**, **ACTIONLINK** and **COLOR** are defined for this.





NAME	INDEX	LENGTH	ALERT	ACTIONLINK	COLOR
Am Segmentende	73	1	▼	-1	green
Batterie Stand [%]	80	8	▼	10	green
Batterieladung aktiv	78	1	▼	-1	green
Betriebsart AUTO	68	1	▼	-1	yellow
Betriebsart HAND	67	1	▼	-1	yellow
Fahrauftrag aktiv	71	1	▼	-1	green
Fahrzeug faehrt	72	1	▼	-1	yellow
Kein Segment	70	1	▼	1	red
Ladeablauf aktiv	77	1	▼	-1	yellow
Laserscanner	66	1	▼	1	red
Notaus	64	1	▼	1	red
Pos zum Laden zu ungena	76	1	▼	-1	red
Position vorhanden	69	1	▼	-1	green
Schwache Batterie	79	1	▼	1	red
Stoerung	65	1	▼	1	red

Figure 30: Model Editor – Status

The name describes the transmitted information. The **Index** indicates the position where the information is located in the receiving telegram. The length specifies the number of digits in the telegram.

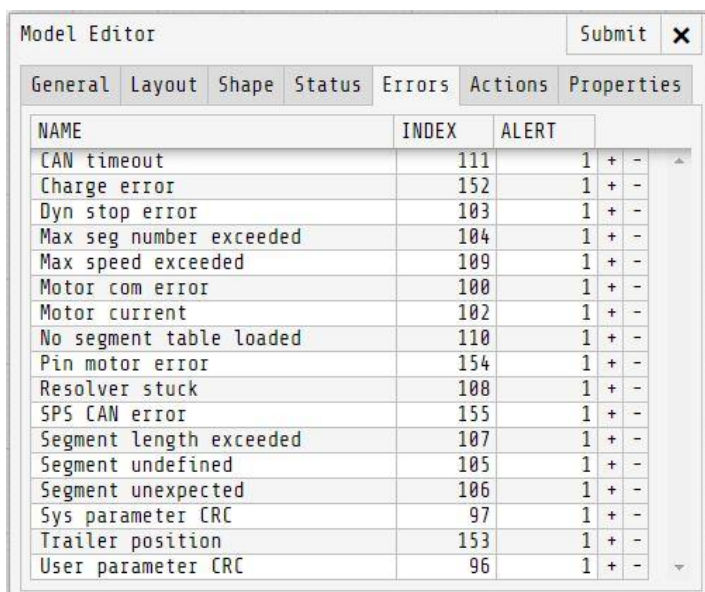
ALERT describes by which information transfer an alarm is set. A set alarm is displayed on the map with a flashing circle around the target and under the target status info in the **Targets** resource table with a colored light. The color of this light and the circle is defined under **COLOR**.

If the **ALERT** status is set to 0, it will always be displayed in red. If the value in the **ALERT** column is set to -1, the transmitted status of the vehicle is not taken into account. With the value 1, the status only lights up when the respective status is received from the vehicle.

In the **ACTIONLINK** column, a **Status** can be linked to a vehicle action, which must be defined in the **ACTIONS** tab.

10.5. Errors

Under **Errors** the names of the errors, the indices and the alert value are defined. The meaning is the same as for the statuses. **Errors** are also defined by the vehicle manufacturer.

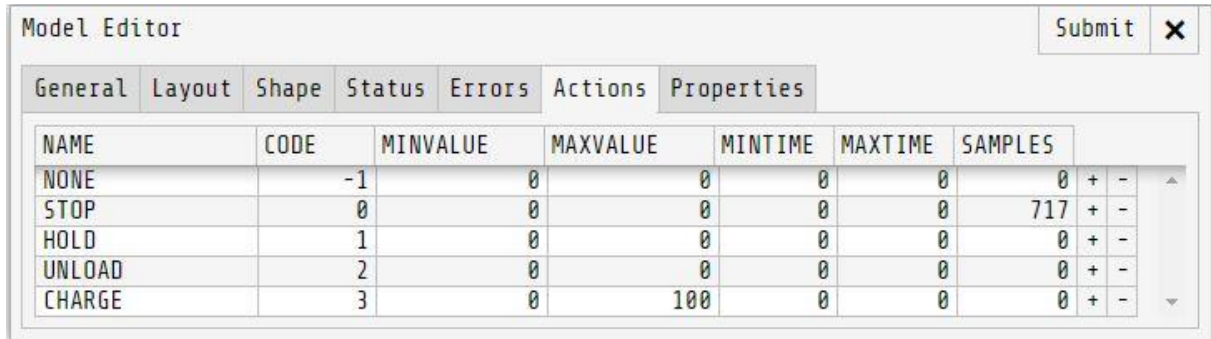


NAME	INDEX	ALERT
CAN timeout	111	1 + -
Charge error	152	1 + -
Dyn stop error	103	1 + -
Max seg number exceeded	104	1 + -
Max speed exceeded	109	1 + -
Motor com error	100	1 + -
Motor current	102	1 + -
No segment table loaded	110	1 + -
Pin motor error	154	1 + -
Resolver stuck	108	1 + -
SPS CAN error	155	1 + -
Segment length exceeded	107	1 + -
Segment undefined	105	1 + -
Segment unexpected	106	1 + -
Sys parameter CRC	97	1 + -
Trailer position	153	1 + -
User parameter CRC	96	1 + -

Figure 31: Model Editor - Errors

10.6. Actions

By defining **Actions**, TransportControl can initiate the execution of physical actions by targets. For each action, a **NAME**, the **CODE** and the minimum and maximum value must be defined.



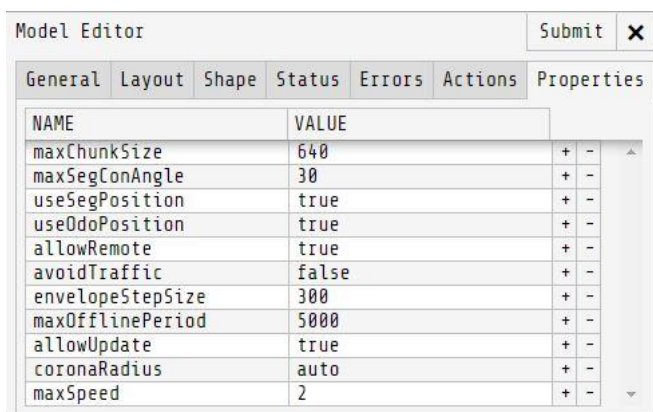
NAME	CODE	MINVALUE	MAXVALUE	MINTIME	MAXTIME	SAMPLES
NONE	-1	0	0	0	0	0
STOP	0	0	0	0	0	717
HOLD	1	0	0	0	0	0
UNLOAD	2	0	0	0	0	0
CHARGE	3	0	100	0	0	0

Figure 32: Model Editor – Actions

The entered actions are available to jobs after confirmation with **Submit** during the editing and creation in the drop-down list. The **MINVALUE** and **MAXVALUE** columns define the minimum and maximum values of the action. The **CODE** is an shortcut for the transmission to the vehicle and is linked to the corresponding action in the vehicle. Each time an action is executed, it is determined how long the target requires for the execution. The times will be saved and the average value will be calculated from them. This average time is included in the calculation of the execution time of the jobs (**Appointment time**). In this way, the calculation of the execution time can be made more accurately.

10.7. Properties

Different characteristics of the **Model** are defined in **Properties**. For this purpose, the desired value is entered in the corresponding line.



NAME	VALUE
maxChunkSize	640
maxSegConAngle	30
useSegPosition	true
useOdoPosition	true
allowRemote	true
avoidTraffic	false
envelopeStepSize	300
maxOfflinePeriod	5000
allowUpdate	true
coronaRadius	auto
maxSpeed	2

Figure 33: Model Editor – Properties

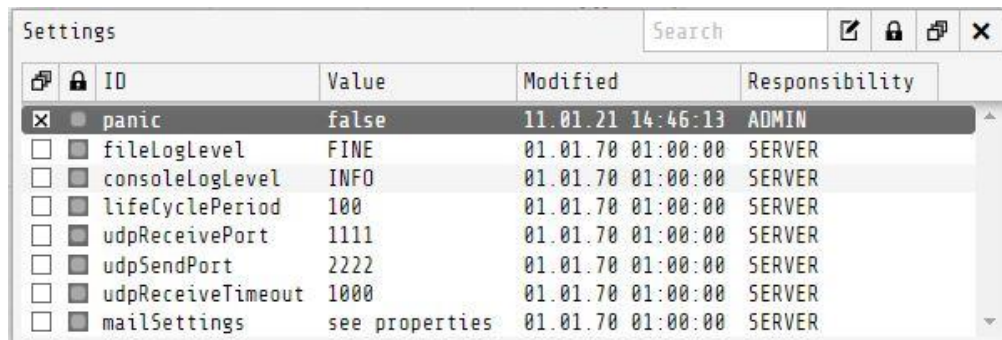
Name	„Value“-Entries	Explanation
maxChunkSize	Numerical value	Describes the size of the payload data of a single update telegram.
maxSegConAngle	0-360	Defines the maximum drivable angle at the connection point of two segments in degrees.

useSegPosition	true, false	Use of position optimization based on segment progress (in case of inaccurate odometry).
useOdoPosition	true, false	Use of the position calculation based on the transmitted data from the vehicle telegram.
allowRemote	true, false	Specifies if targets based on this model are allowed to be controlled remotely.
avoidTraffic	true, false	Only unoccupied segments are taken into account in route finding (intended for demonstration purposes only).
envelopeStepSize	Numerical value	Allows you to set the step size of a calculation step for the envelope curve and is specified in mm.
maxOfflinePeriod	Numerical value	If TransportControl does not receive a telegram from a target within this time span (in m/s), the target is considered offline.
allowUpdate	true, false	Allows the transmission of updates via a special update telegram (if supported by the target).
coronaRadius	Numerical value	Defines the size of the red blinking circle around a target on the map. The size is given in millimeters.
maxSpeed	Numerical value	Defines the maximum speed in m/s in TransportControl for this vehicle type.

Table 2: Model properties, value entries and effects


11. Settings

In the **Settings** resource table, settings can be made without restarting TransportControl.



ID	Value	Modified	Responsibility
<input checked="" type="checkbox"/> panic	false	11.01.21 14:46:13	ADMIN
<input type="checkbox"/> fileLogLevel	FINE	01.01.70 01:00:00	SERVER
<input type="checkbox"/> consoleLogLevel	INFO	01.01.70 01:00:00	SERVER
<input type="checkbox"/> lifeCyclePeriod	100	01.01.70 01:00:00	SERVER
<input type="checkbox"/> udpReceivePort	1111	01.01.70 01:00:00	SERVER
<input type="checkbox"/> udpSendPort	2222	01.01.70 01:00:00	SERVER
<input type="checkbox"/> udpReceiveTimeout	1000	01.01.70 01:00:00	SERVER
<input type="checkbox"/> mailSettings	see properties	01.01.70 01:00:00	SERVER

Figure 34: Resource table Settings


Via the selection of the table entries and the  button the settings can be edited. These include the PANIC button, the log level, the core loop time, the UDP ports and the time-out of the UDP reception.

12. Users

A user who is logged into TransportControl and has the correct authorization is always required to retrieve, create, modify, or delete resources. Users authenticate themselves by entering a **Username** and the corresponding **Password**.

Individual authorizations can be assigned to each user. These authorizations are based on the resource classes of TransportControl.



In the resource table all users are displayed with their ID, the username. The  column indicates which users are logged in. The **Influence** column provides information about the number of resources currently managed by this user. To protect the system against accidental or intentional overload, each user can be assigned a maximum of the resources that they can manage in the user administration.

















Users							Search							
		ID		Influence	IP	Last request	Remote	Modified	Responsibility					
<input type="checkbox"/>		New_User		2507	0:0:0:0:0:0:1	03.11.20 10:02:56	NONE	03.11.20 10:04:46	GSF_AMI					
<input type="checkbox"/>		GSF_AMI		58	0:0:0:0:0:0:1	11.01.21 10:28:09	NONE	01.01.70 01:00:00	SERVER					
<input type="checkbox"/>		ADMIN		28	0:0:0:0:0:0:1	11.01.21 14:46:13	NONE	11.01.21 10:28:09	GSF_AMI					

Figure 35: Resource table users


Users who are still registered in the general resource management, which means that their **Influence** value is greater than zero, **cannot be deleted from the system**.

The IP address from which the server was last accessed is displayed in the **IP** column. It is also recorded for each user when he last had contact with the server. In addition, it is displayed if a target is currently being remotely controlled by the user. The responsibility indicates who adjusted the user last. The user ID can no longer be adjusted after a user has been created, as otherwise it would no longer be possible to trace the entries in the **Responsibility** field of each resource table.

Via the buttons  and  a user profile can be downloaded or uploaded.

The  button opens the **User Editor**, which can be used to create new users or edit existing users. The **ID** is unchangeable after creation.

The Authentication field defines the method with which the user logs in. The Authentication field defines the method with which the user logs in. This determines if the user table is taken from the database used by TransportControl or if an associated LDAP server is used to authenticate the user instead.

A password is finally defined for the user, which has at least ten characters. Via  the password can be displayed in plain text.

User Editor

Submit

X

General

Permissions

ID

ADMIN

Authentication

INTERNAL

Password

Leave blank to keep


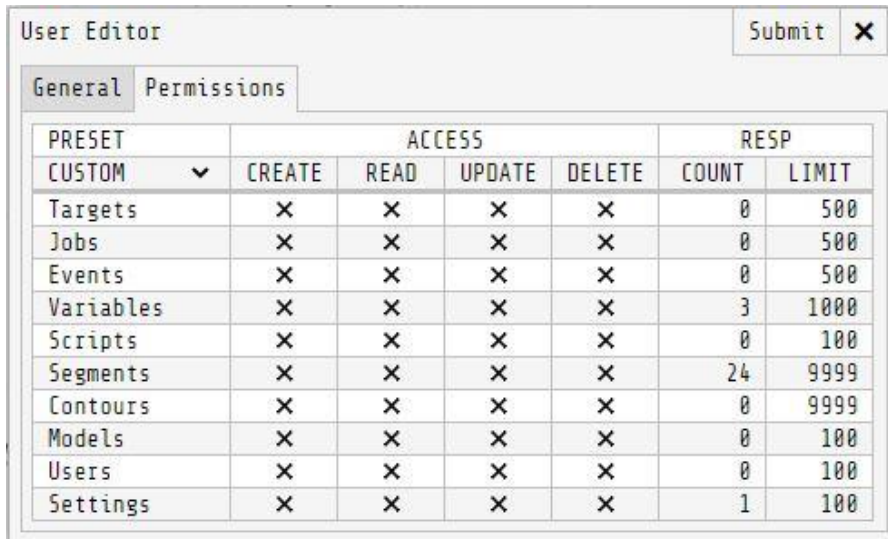


Figure 36: User Editor – General

Authorizations are defined in the **Permissions** tab. A cross is a granted permission, an empty field represents no permission. The terms **CREATE**, **READ**, **UPDATE** and **DELETE** in the **ACCESS** area are based on the SQL standard.

The **COUNT** and **LIMIT** entries found in the **RESPONSIBILITY** area reflect how many resources of a certain type are currently controlled by the user and how many he is allowed to control at maximum. To allow the user to create, modify and delete resources, the **LIMIT** must be greater than zero.




The 'User Editor' window has a 'Permissions' tab. It contains a table with columns: PRESET, CUSTOM (with a dropdown arrow), ACCESS (subdivided into CREATE, READ, UPDATE, DELETE), and RESP (subdivided into COUNT and LIMIT). The table lists permissions for various resources, all marked with 'X' in the ACCESS columns.

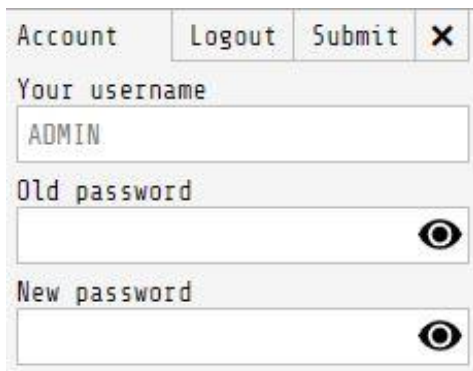
PRESET	CUSTOM	CREATE	READ	UPDATE	DELETE	COUNT	LIMIT
Targets		X	X	X	X	0	500
Jobs		X	X	X	X	0	500
Events		X	X	X	X	0	500
Variables		X	X	X	X	3	1000
Scripts		X	X	X	X	0	100
Segments		X	X	X	X	24	9999
Contours		X	X	X	X	0	9999
Models		X	X	X	X	0	100
Users		X	X	X	X	0	100
Settings		X	X	X	X	1	100

Figure 37: User Editor – Permissions

In addition to the manual selection of all authorizations, it is possible to use one of the various authorization levels from the **PRESETS** drop-down menu. The crosses within the authorization matrix are set according to the selection. By a manual adjustment after selecting an authorization level, the authorization level changes back to **CUSTOM**. For each resource, the user **ID** of the last editor is assigned in the **Responsibility** field. Users can be deleted after the resource responsibility is assigned to other users.

12.1. Change password

You can change your password after logging in by clicking the main menu button . The window which appears allows you to log out and change your password. For this purpose, the current and the new password must be entered and the information confirmed with **Submit**. The **Your Username** field is automatically filled with the logged in username.

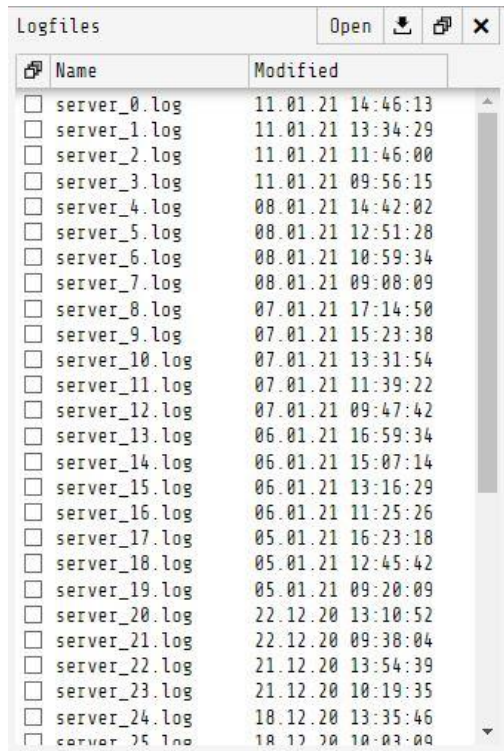


The 'Account' window has buttons for 'Logout', 'Submit', and a close button (X). It contains three input fields: 'Your username' (pre-filled with 'ADMIN'), 'Old password', and 'New password'. Each password field has a toggle icon (an eye) to switch between visible and hidden states.

Figure 38: Account window

13. Logfiles

TransportControl writes **Logfiles** during runtime and stores them on the server in the installation folder. If the currently written file has grown to a size of 1 MB, a new file is created. A maximum of 1000 **Logfiles** will be created before the oldest file is overwritten by new entries.



Name	Modified
server_0.log	11.01.21 14:46:13
server_1.log	11.01.21 13:34:29
server_2.log	11.01.21 11:46:00
server_3.log	11.01.21 09:56:15
server_4.log	08.01.21 14:42:02
server_5.log	08.01.21 12:51:28
server_6.log	08.01.21 10:59:34
server_7.log	08.01.21 09:08:09
server_8.log	07.01.21 17:14:50
server_9.log	07.01.21 15:23:38
server_10.log	07.01.21 13:31:54
server_11.log	07.01.21 11:39:22
server_12.log	07.01.21 09:47:42
server_13.log	06.01.21 16:59:34
server_14.log	06.01.21 15:07:14
server_15.log	06.01.21 13:16:29
server_16.log	06.01.21 11:25:26
server_17.log	05.01.21 16:23:18
server_18.log	05.01.21 12:45:42
server_19.log	05.01.21 09:20:09
server_20.log	22.12.20 13:10:52
server_21.log	22.12.20 09:38:04
server_22.log	21.12.20 13:54:39
server_23.log	21.12.20 10:19:35
server_24.log	18.12.20 13:35:46
server_25.log	18.12.20 10:03:09


Figure 39: Resource table Logfiles

The information that is written to the **Logfiles** depends on the set log level. The log level can be changed at any time via the Main UI in the **Settings** resource table with the corresponding authorization. Each finer level includes all coarser levels.

The log levels are defined as follows:

INFO	Server info, valid telegram inputs, script logging, remote/update
FINE	Logging in and logging out of targets + users, radio interference + errors of all targets, all changes to resources
FINER	Status messages of all targets
FINEST	Motion data of all targets: segment/point/speed/X/Y/angle

Table 3: Log level scope


Logfiles can be opened directly in the browser via the **Open** button in the resource table or downloaded via the  Button. The created Logfiles are stored directly in the installation folder of TransportControl. These can be retrieved under the path .../transportcontrol/logfiles. The **Logfiles** are simple text files and can theoretically be read with any editor. Each line of the file documents an event within TransportControl and consists of date, time, log level and the message itself.



14. PANIC

With the PANIC button, users with the permission to change the resource **Settings** can set a global stop. After pressing the PANIC button, the button is colored yellow and all vehicles stop until the PANIC button is pressed again. If the PANIC button is colored red, a critical error has occurred within TransportControls. Until the error is corrected, all vehicles will stop. Clicking the red PANIC button opens a window with the description of the occurred error and the **Clear** and **Cancel** buttons. By selecting the error, a more detailed description appears. Clear resets all error messages. If the error still exists, the PANIC button remains red. Clicking **Cancel** closes the window without resetting the error.

15. Account

On the right of the PANIC button is the Account  button. This allows the currently logged in user to set a new password or log out of the system.

We are happy to receive comments, suggestions or hints for improving the operator's manual by e-mail (contact@movizon.de).

