RAILROAD & CO.®

TrainController™

Change Description

Version 7

October, 2008

Contact: Freiwald Software
Kreuzberg 16 B
D-85658 Egmating, Germany
e-mail: contact@freiwald.com
http://www.freiwald.com

Table of Contents

About this Document

RAILROAD & CO. is the leading product line of computer programs for digitally or conventionally controlled model railroads. It contains the following members:

- **TrainController™ 7** is the world's leading software for model railroad computer control.
- **TrainProgrammer™** is the program, which makes programming of DCC decoders as simple as a few clicks with your mouse.
- **+Net™** is a module, that allows to control your layout with a network of several computers running **TrainController™**.
- **+4DSound™** is a module, that recreates realistic spatial sound effects for each model railroad layout controlled by **TrainController™** without the need to install on-board sound into each decoder.
- **+SmartHand™** is the world's premium handheld railroad control system designed for computer controlled model railroads.

TrainController™ 7
Differences to TrainController™ 5

TrainController™ 7 describes the latest generation of **TrainController™** software. **TrainController™ 7** provides so many new features, outstanding improvements and changes, that the terms “new version” or less than ever “upgrade” to an existing version are not appropriate. **TrainController™ 7** concentrates a number of innovations, that are otherwise spread to several upgrade versions of the same product. **TrainController™ 7** is not only a single new product, it is a complete line of new products.

In fact **TrainController™ 7** is offered in three variants:

- **TrainController™ 7 Bronze** provides a low-cost entry into computer controlled model railroads. It is primarily designed for users with small and medium size layouts and average requirements. Novice users, who do not know **TrainController™**, may consider to do their first steps with **TrainController™ 7 Bronze**, too. The reduced functionality of this variant makes it easier to identify and to learn the basic functions of **TrainController™**.
- **TrainController™ 7 Silver** is the successor of the established and well-known version **TrainController™ 5**. It is primarily designed for advanced users with upmarket requirements and owners of layouts of all sizes. While **TrainController™ 5** was already outstanding with regard to functionality, ease of use and quality the im-

provements introduced in **TrainController™ 7** strengthen the leading position of **TrainController™ 7 Silver** compared to the available competitors.

- **TrainController™ 7 Gold** is the flagship of the **TrainController™** family and a class of its own. **TrainController™ 7 Gold** is primarily designed for users with supreme requirements, who want to operate their layout like the real professionals. While **TrainController™ 7 Silver** is already able to operate even very large layouts, **TrainController™ 7 Gold** provides much more convenience, efficiency and security for design and operation – especially for larger layouts.

This document provides an overview of the features, which are unique in **TrainController™ 7** and which distinguish **TrainController™ 7** from **TrainController™ 5**. It is mainly intended for users, who know **TrainController™ 5**, and who want to learn about the differences between **TrainController™ 7** and **5**. It is assumed, that users of **TrainController™ 5** are primarily interested in the features of **TrainController™ 7 Silver** and **Gold**. Since the functionality of **TrainController™ 7 Bronze** is somewhat limited compared to **TrainController™ 5**, no user of **TrainController™ 5** will probably consider to change to **TrainController™ 7 Bronze**. For this reason this document focuses on the differences between **TrainController™ 5** on one side and **TrainController™ 7 Silver** and **Gold** on the other.

The features of **TrainController™ 7 Bronze** are described in a separate document.

It is assumed, that the reader is familiar with **TrainController™ 5.8** and the **TrainController™ 5.8 Users Guide**. New users of **TrainController™ 7** should focus on the **TrainController™ 7 Users Guide** rather than this document.

The numbers of the particular chapters and sections of this document are inherited from the concerning chapters and sections of the **TrainController™ 5.8 Users Guide**. This allows readers, who are familiar with that Users Guide, to implicate the contents of both documents.

All text sections, that describe features of **TrainController™ 7 Gold**, which are not provided by **TrainController™ 7 Silver**, are marked with a specific marking on the left side of the text in the same way as this section. Contents marked in this way do not apply to **TrainController™ 7 Silver**. Users of this program version or readers only interested in **TrainController™ 7 Silver** can safely ignore these contents.

The Differences at a Glance

In the following all differences, that apply to **TrainController™ 7 Silver** and **Gold** are listed:

Misc:

1. Official support of Microsoft Windows Vista.
2. Completely new design of the user interface with selection of different general user interface styles (e.g. Microsoft Office, Visual Studio, etc.) See page 30.
3. Improved window management. See page 30.
4. It is possible to arrange more than one window in the background of the main window. See page 30.
5. Standardization of the layout of the particular windows. See page 12.
6. The content of menus and tool bars can be customized (see page 33).
7. Keyboard accelerators can be changed (see page 33).
8. Each file can be reloaded with default window settings to enable repair of disarranged window arrangements.
9. Files are compressed to save disk space.
10. Improved support for lefties: the software now responds accordingly, if the left and right mouse buttons are swapped.
11. There are some other differences, that are to be taken into account, when migrating existing data files from **TrainController™ 5** to **TrainController™ 7**. These differences are outlined beginning at page 125.
12. A tip of the day window provides useful tips.

Switchboard:

13. Extended customization options.
14. Six highlighting styles can be applied generally or differing for plain tracks vs. turnouts, occupied vs. non-occupied track symbols as well as track symbols in active routes. The possibilities to apply different styles and colors to different highlighting aspects provide virtually countless layout combinations and allow reproduction of almost each highlighting style of prototypical plug panels.
15. Images are automatically scaled, if the symbol size changes.
16. It is possible, to specify colours for the background and frame of text elements. Text elements can be individually resized and aligned within their boundaries.
17. Highlighting of occupied track elements can be controlled with different modes and according to different colour schemes. See page 34.
18. Routes, that are automatically created by the automatic creation of the block diagram, can be optionally completely highlighted now. See page 34.

19. Active routes can be highlighted according to different colour schemes. See page 34.
20. The customisation options of each switchboard can be reset to factory defaults.
21. The status of switchboard objects with more than two states (e.g. three or four aspect signals, three-way turnouts, double slip switches) can be set directly with the context menu attached to the right mouse button

Train Window:

22. Completely new design for the train window. See page 39.
23. The throttle can be arranged now with the zero position in the middle or on the left.
24. The directional effect of the throttle can be set to be **layout** or **train** oriented. See page 39.
25. The effect of the throttle control to the acceleration of trains can be set to different modes (speed, power or no momentum). See page 39.
26. The number of steps of the throttle and brake control for control with the keyboard or mouse wheel can be adjusted individually now. A special setting sets the speed steps of the throttle control to the number of speed steps of the physical decoder.
27. The interaction between the throttle and brake control has been improved. Dragging the brake control automatically sets the throttle to zero and vice versa.
28. The size of the train window can be changed in an arbitrary manner.
29. It is possible to define and to restore an optimal size for train windows.
30. New customization options for the train window allow customization of colors and appearance of the speedometer, odometer, throttle, brake and other controls.
31. The customization options of each train window can be reset to factory defaults.
32. Improved image editor for train function symbols.
33. The symbol or the name of the selected train is displayed in the train window.

Train List and Train Management:

34. The train list has been replaced by a much more powerful train grid.
35. The train grid displays additional information for each train such as current block, currently executed schedule, etc.
36. The train grid supports multiple selection now of trains, but also of blocks, schedules etc.
37. It is possible now to assign a list of operations, which shall be executed by a certain auxiliary engine function. Prior creation of a macro is not necessary anymore. If several auxiliary functions shall execute the same sequence of operations, then it is still possible to create macros for this purpose and to assign these macros as a member to such lists.

Dispatcher and Automatic Operation:

38. Completely new design of the dispatcher window.
39. The display intensity of blocks and routes, that do not belong to the selected schedule can be dimmed. See page 82.
40. Highlighting of occupied or active routes can be controlled with different policies and according to different color schemes. See page 82.
41. The design of the dispatcher windows can be reset to factory defaults.
42. The routes contained in each block diagram are now additionally listed in a separate list.
43. Routes can be operated manually by mouse click via the dispatcher window.
44. The assignment of digital addresses to feedback indicator symbols can be assisted by a search run, which scans the available addresses for occupied sensors.
45. If a train is assigned to a block by drag & drop, then the orientation of the train is automatically derived from the position of the mouse within the rectangle of the block. The orientation of the train can additionally be swapped by drag & drop, too.
46. Schedules can be started by dragging a train (e.g. from a switchboard) to the entry of a schedule in the schedule list of the dispatcher window. This is useful for people who desire to work only with the list part of the dispatcher window and who do not want to spend screen space for the diagram part of the dispatcher window.
47. Revised arrangement of the views in the dispatcher window. The schedule view, for example, additionally displays the blocks and routes contained in each schedule in separate lists.
48. For each block or route selected in the dispatcher window it is possible to view the details of this block or route in a separate detail view. This detail view also allows to simulate the indicators assigned to the block or route.
49. The internal calculation routines of the **AutoTrain** feature have been optimised and improved considerably. While previous versions of **TrainController™** sometimes needed minutes to find a path from the specified start to the destination blocks, assumed they found a path at all, **TrainController™ 7** often finds suitable paths even on large or complex layouts within seconds or even fractions of a second. In previous versions **AutoTrain** was not even able to find a short path between two blocks, if the track diagram of the layout was large or complex and there were turnouts on the path between both blocks, which led to other (complex) parts of the layout. Such short parts are now usually found within a fraction of a second, even on the largest and most complex layouts.
50. In previous versions of **TrainController™** it was possible to prevent **AutoTrain** from searching “endlessly” by specifying a maximum number of blocks, which may be included into the found path (maximum path length). With that approach it was not possible to create longer paths with **AutoTrain** without running the risk of starting an endless search. In **TrainController™ 7** a maximum search time is spe-

cified rather than a maximum path length. This allows first to let **AutoTrain** search for much longer paths than previously and this ensures second, that **AutoTrain** finds at least a major part of the suitable paths even in the rare cases, where this search time expires.

51. It is possible to exclude routes from or to include routes into the search for a path performed with the **AutoTrain** tool bar. In previous versions this could only done with blocks (see page 75).
52. In addition to the 'A' key it is also possible to initiate and terminate **AutoTrain** by Drag & Drop with an appropriate menu command or toolbar button (see page 74).
53. **Run with Interlocking** is a new, very handy method to run trains under full protection, routing and signaling. Just put a locomotive on the track and call the menu command **Run with Interlocking**. The train will immediately start to move, provided that the route ahead is clear. It will then select an appropriate path and continue to travel, until it reaches a dead end or until the path ahead is blocked for another reason. At a dead end it can reverse automatically, if desired, and continue to travel to the opposite direction. It is either possible to allow the computer to select and activate all routes requested by the train automatically; it is also possible to leave this to the human operator. See page 25.
54. Different driving modes (e.g. manual vs. automatic train control) can be specified individually for each particular engine. The mode can be changed at any time during operation. The driving mode is applied, when a train containing this engine is running with interlocking. Optionally it is also possible to apply the individual driving mode to train runs controlled by **AutoTrain™** or schedules, too. In the latter case the driving mode of the engine overrides the driving mode of the schedule and allows the same schedule to be used for different trains with different driving modes. See page 73.
55. The new block editor makes it much easier and much more intuitive to create and arrange the indicators, signals and markers in a block (see page 52).
56. The intuitive display of indicators and markers in a block provided by the block editor is also used by the traffic control and by the detail view of the dispatcher window.
57. It is possible to limit the effect of all brake and stop markers to certain conditions. In this way it is possible to define different stop locations in the same block for different operational situations.
58. Shifted brake and stop markers extend the existing concept of virtual contacts. They combine the effect of brake and stop indicators with the characteristics of virtual contacts, but they are much simpler and much more intuitively configured than those. With shifted brake and stop markers it needs only a few mouse clicks to shift the location, where trains stop or begin to brake off the boundary between two occupancy sections. See page 63.

59. Speed limits are not raised, before the block or route is released, to which the limit applies. Trains always run at the minimum of all speed limits valid for all currently reserved blocks and routes.
60. The memory of indicators can now be reset together with another indicator, i.e. when another indicator is turned off.
61. For reasons of simplicity all links in block diagrams are now replaced by routes. This provides additional useful possibilities. Among others the highlighting of the path currently used by a train is improved. There is also no need anymore to distinguish between 'links' and 'routes', which simplifies the use of the software.
62. Blocks can be locked for a certain direction of travel (see page 51).
63. The maximum time limit of the system operation "Delay" has been raised from 10 seconds to 24 hours.
64. It is possible now to assign a list of operations, which shall be executed at the start or termination of a schedule or during a schedule. Prior creation of a macro is not necessary anymore. If several schedules shall execute the same sequence of operations, then it is still possible to create macros for this purpose and to assign these macros as a member to such lists.
65. The **Simulator** allows automatic simulation of model railroad operation without connection to an actual model railroad layout and without requiring human intervention (see page 86).

Timetable / Clock:

66. The timetable has been improved by several features and is displayed in a separate window now.

Turntable:

67. New turntable operation "go to track with marker (house)".

Traffic Control:

68. The layout of the traffic control has been completely revised and is much more intuitive now. See Page 84.

Message Window:

69. The logging of messages can be stopped at any time during operation. This is useful for analysis of messages during running operation.
70. The message window shows an additional column, which displays the name of the train, which each message belongs to, if any.

Inspector / Explorer:

71. If a folder in an Explorer window is currently selected, then the according Inspector window shows the statistics (i.e. number of turnouts, trains, etc.) of all elements stored in this folder or stored in sub folders of this folder. By selecting the root folder the statistics of the complete layout are displayed.

+SmartHand:

72. The **+SmartHand™** window has been integrated into the common window management. It can be docked, floated or treated in any manner now like all other windows.
73. The directional effect of the throttle can be set to be **layout** or **train** oriented. See page 39.
74. The effect of the throttle to the acceleration of trains can be set to different modes (speed, power or no momentum). See page 39.
75. The number of steps of the throttle can be adjusted individually now. A special setting sets the speed steps of the throttle control to the number of speed steps of the physical decoder.
76. A new optional ballistic mode allows more significant speed changes by turning the throttle knob quickly.
77. Running with Interlocking (see page 25) can be initiated and performed with **+SmartHand™** handhelds, too.

+Net:

78. With a new option it is possible to flip the orientation of a block towards the network. The right or bottom side of a flipped block is reported as left or top side to the network and vice versa. If a train is assigned to such block with orientation to the right, then this is reported as orientation to the left to other computers in the network and vice versa.

Hardware and Digital Systems:

79. Lenz Version 3.6: Support of locomotive functions F13 – F28
80. Tams EasyControl: Support of locomotive functions F9 – F14
81. Official support for Digitrax PR3
82. Official support für Uhlenbrock LocoNet Interfaces

The following differences only apply to **TrainController™ 7 Gold**:

Misc:

83. Wildcards in object names. See **Help** menu.
84. A descriptive comment can be added to each object. This comment is displayed in the tool tip window, when the mouse is moved to the element. The comment is also included in the printout of the object details.
85. A new menu command **Lock Start** prevents **TrainController™ 7** from terminating an emergency stop state and restarting all interrupted processes, when the start button of the digital system is pressed. This option is useful, if a powered digital system is required to resolve a certain emergency situation, and if it is not desired, that **TrainController™ 7** continues its processing, while the emergency situation is not resolved. Setting this option allows to start the digital system, while **TrainController™ 7** remains stopped.

Switchboard:

86. Switchboard symbols can be displayed in five different sizes ranging from 12x12 to 28x28 pixels per symbol / switchboard cell.
87. Additional track symbols: space saving turnout elements and adequate connecting tracks and crossing symbols. These new track symbols do not only allow space saving arrangement of turnouts, but also reproduction of certain prototypical control panel layouts (see page 35).
88. The name of the associated block can be displayed in block diagrams or the switchboard, too, when edit mode is turned off.
89. It is possible, to override the default colours for the background and frame of text elements by individual settings for each particular text element.
90. Several switchboards can be linked together using connector symbols. These connector symbols are also used for automatic linkage of several calculated block diagrams (see also entries 113, 113 and 116 unterhalb). By double clicking to these connector symbols it is possible to navigate from one switchboard to an adjacent switchboard.
91. Track elements in switchboard windows can be colorized individually. The **Tools** menu provides additional commands for coloring of individual tracks or for coloring of contiguous track sections.
92. It is possible to insert an empty new line or column at any position into the switchboard with one menu command. The lines and columns right/below the inserted line/column are shifted accordingly. This action can be undone, too. In a similar way it is possible to delete a complete line or column from the switchboard.
93. It is possible to create custom switchboard symbols for signals, push buttons, on/off switches, toggle switches, routes, feedback indicators, flagmen and virtual contacts with an integrated bitmap editor and to assign such custom symbols individually to each according switchboard object. Custom switchboard symbols can be transferred between different data files by export and import.

94. It is possible to create self-provided, inoperable switchboard symbols with an integrated bitmap editor (see page 36).
95. The second digital address of switchboard objects with more than two states (e.g. three-way turnouts or four-aspect signals) can be specified independently from the first address.
96. Mini block symbols can be used to represent blocks in diagonal track sections.
97. Individual blocks can be hidden in the switchboard, when edit mode is turned off.
98. The display of block signals in blocks can be turned on and off individually, too, i.e. on a per-block base.

Train Window:

99. **TrainController™ 7** supports the operation of train functions controlled by additional function-only decoders without the need to create an artificial multiple-unit as required in **TrainController™ 5**.
100. It is possible to specify an individual maintenance interval for each engine or car and an optional operation, that is automatically executed, when the maintenance interval expires (see page 98).

Train List and Train Management:

101. Train groups can be optionally defined to exclude all vehicles listed therein. Vehicles are contained in such train group, if they are not listed in this group (see page 95).
102. The train management has been completely revised. It is now possible to define cars and to arrange train sets (multiple units, consists) at any time during operation (see page 88).
103. Vehicles can be joined to train sets automatically by means of Operations. Train sets can be separated automatically by Operations, too (see page 94).
104. A new schedule rule allows trains to enter reserved destination blocks to join vehicles, that are already located there, to form a new train set (see page 92).
105. Multiple units can be operated with the throttle of the digital system. Multiple units can be created and dissolved with minimum human intervention. Just turning the throttle knob for one participating engine is already sufficient (see page 92)
106. It is possible to specify an individual length for each specific car. This is taken into account for the calculation of the total length of each train set. By adding or removing such cars from/to train sets during operation, the total length of each train can changes automatically, which again is taken into account for stops at the middle of platforms or the extended train guidance based on train length (see 111).
107. For realistic simulation of train tonnage it is possible to specify the full weight and the empty weight for each car. Cars can be loaded and unloaded manually or automatically at any time during operation. The currently selected weight (load condi-

tion) of each car is applied to the calculation of the maximum speed or acceleration momentum of affected train sets (see page 91).

108. It is possible to specify individual contact spots for each specific car and both directions. This is automatically taken into account for proper calculation of braking ramps and distant markers, when a train set is currently being pushed.
109. Forwarding of functions can be turned on or off for each train set at any time during operation (see page 91).
110. In addition to the five variants to measure the speed profile of an engine **TrainController™ 7** provides two procedures to trim the brake compensation of each engine efficiently (see page 43).
111. New train operations allow to run the current train with interlocking or to terminate the current schedule of a train. These operations can be called automatically by contact indicators, by brake, stop, speed or action markers or by macros.
112. Temporary speed limits can be set with a new train operation (see page 71).

Dispatcher and Automatic Operation:

113. The name of the blocks can be displayed, too, when edit mode is turned off.
114. It is possible now to create more than one block diagram. See page 46.
115. The automatic calculation of the block diagram can involve more than one switchboard window. See page 46. Due to this feature **TrainController™ 7 Gold** is much better suited for medium and large layouts than other versions of **TrainController™** (including the **Silver** version). where the automatic calculation of the block diagram is always limited to a single switchboard.
116. It is possible now to open more than one dispatcher windows at the same time. This is useful if you want to monitor different block diagrams simultaneously.
117. The **Passenger Ride** command of the **View** menu causes the dispatcher window to follow the selected train as it moves across the layout. The block, in which the train is located at a time, will be automatically highlighted and displayed in the dispatcher window. If the train moves to another block diagram, then the display changes to this diagram, too.
118. Connector symbols allow to link several switchboard windows or dispatcher windows to each other. See page 50. Connector symbols can also be used to create hidden track connections within the same switchboard or block diagram.
119. With a new option it is possible to exclude those routes from the calculation of block diagrams, that contain too much turnouts. If two routes between the same two blocks contain a different number of turnouts and this difference exceeds a certain preset value, then the route with the higher number of turnouts is ignored. By using this option the calculated block diagrams will contain only routes with a minimum or accordingly higher number of turnouts.
120. New speed markers provide more control of the location, where speed limits of the subsequent block are applied. See page 58

121. New action markers allow to trigger operations easily at any location in a block without affecting the speed of the passing train. See page 59.
122. The effect of all brake, stop, speed and action markers can be limited to specific trains. In this way it is very simple to let passenger and freight trains stop at different positions. Many other useful applications can be easily achieved, too. See page 66.
123. The effect of all brake, stop, speed and action markers can be limited to specific schedules. In this way it is very simple to let the same train stop at different positions depending on the currently executed schedule. Many other useful applications can be easily accomplished, too. See page 66.
124. It is possible to specify differing brake and stop markers for scheduled stops and for unscheduled stops in the same block. This can be used, for example, to let the same train perform scheduled stops in the middle of a platform and perform unscheduled stops near the block signal at the end of the block, for example. See page 67.
125. It is possible now to assign indicators to turnouts for occupancy indication of all routes, that are using the concerning turnouts (see page 111).
126. It is now possible to specify a separate set of schedule rules for AutoTrain runs. These schedule rules work in the same way as schedule rules for regular schedules. They can be changed outside edit mode, however, and each change affects all AutoTrain runs, which are initiated later.
127. **AutoTrain** can be called by operations of other objects. It is in particular possible to trigger **AutoTrain** by start and destination keys; even from external control panels. This allows for automatic train runs on point to point connections without the need to create schedules in advance. (see page 76).
128. Individual schedule rules can be applied to all schedules on request.
129. New schedule rules prevent schedules from reserving occupied routes and blocks without the need to specify extra conditions for the blocks or routes. In other versions of **TrainController™** it is only possible to prevent trains from running into such blocks or routes. To prevent reservation, too, it is necessary there to specify appropriate conditions.
130. An additional *smart* mode can be optionally applied as a new rule to the release of routes in schedules: in this mode passed routes with own occupancy indication are released, when they are no longer reported as occupied. Routes without own occupancy indication are released, when the train reaches a stop marker of a subsequent block. In this way the smart mode automatically selects the release policy, which is optimal for the particular route.
131. Routes, that were already activated prior to reservation by a schedule, are optionally deactivated automatically upon termination of the schedule, if desired. In other versions of **TrainController™** such routes remain always activated and must be turned off explicitly. This is controlled by a new schedule rule.

132. A new schedule rule keeps routes and blocks, that could not be released during the normal run of the schedule, reserved upon termination of the schedule, if desired. These blocks or routes are automatically released later, when this is possible. In other versions of **TrainController™** all routes and blocks requested by a schedule and different from the current block of the train are always released upon termination of the schedule.
133. A new optional schedule rule causes schedules to select always that route among several routes between the same two blocks, that contains the smallest number of turnouts. This rule is activated by default for new schedules to prevent trains from passing undesired crossovers.
134. With a specific schedule rule it is possible to specify a schedule watchdog. This is the maximum time period between activation of two indicators. If no indicator is triggered within the specified period of time and the train is set to run at non zero speed, then it is assumed, that the train got stuck. In such cases appropriate error information is displayed on the screen (see page 114) .
135. A specific schedule rule provides limited aberration protection. If a train running under control of such schedule is detected in an unexpected block, then appropriate measures are automatically taken by the software (see page 114) .
136. With a specific schedule rule it is possible to specify, that always that path is selected, that contains routes or blocks, which have been visited by the train under control of this schedule the longest time ago ("oldest" block or routes). This option can be used to implement systematic track cleaning trains (see page 114).
137. It is possible to specify a start delay for each schedule, which is applied at the beginning of each schedule and after each stop of a train in a schedule. This delay specifies the time span, which shall take place between clearance of a track section ahead of the train and before the train is set in motion. This time span simulates the response time of the engineer.
138. In addition to the global start delay described above, which applies to all stops, scheduled and unscheduled, in equal measure, it is also possible to specify an individual delay for each scheduled stop. Such delay is applied after a scheduled stop and execution of the associated operations and before the train is set in motion. This time span can be utilized to perform additional operations (e.g. playing an announcement, the noise of closing doors or the whistle of the conductor) after a scheduled stop ended and before the train is set in motion (see page71).
139. It is possible to prevent certain schedules (e.g. schedules solely used as successors of other schedules) from being listed in the dispatcher window, when edit mode is turned off (See page 83).
140. Blocks can be defined to be permanently unidirectional. Such blocks can only be passed in a certain direction of travel. Unlike temporary entry locks (see list entry 62 above), which cause a similar effect, this setting is permanently valid and can only be changed in edit mode (see page 51).

141. Blocks, routes, schedules, trains, turnouts and other objects can be decommissioned and excluded from operation at any time during operation (see page 104).
142. Train guidance based on train length: Each train can be prevented from going to destination blocks, that are shorter than the train (see page 111).
143. Train guidance based on train length: Each train can be prevented from stopping in blocks, that are shorter than the train. (see page 111)
144. Train guidance based on train length: Each train can be prompted to prefer the shortest destination block, which is long enough to store the train (see page 111).
145. A new schedule rule prevents blocks and routes from being released during a running schedule, if the train length indicates, that the train does not fit completely into subsequent blocks.
146. Extended Train Guidance System: Each train can be forced to start a schedule in a certain direction, i.e. forward vs. backward or pulling vs. pushing, respectively. It is also possible to specify, that trains can only be started, if they maintain their current direction of travel (see page 112).
147. Extended options for the selection of trains for schedule successors and schedule selections: schedule successors or schedule selections can be started with specific trains. The option to start a schedule successor with the oldest train can also be applied, if a train change is desired (see page 77).
148. In cases, where control of running trains is passed from a schedule to a successor schedule without stopping the train, a new schedule option causes allocation of blocks and routes of the successor schedule already, when the train enters the second last block of this schedule. Usually and in other versions of **TrainController™** this allocation is not performed, before the train enters the destination block of the schedule. This option allows for a more fluent change of control between schedules and improved calculation of block signals during this change.
149. Schedule sequences allow sequencing of single schedules, which is usually more flexible then the static chaining of schedules as successors (see page 77).
150. Each schedule can be optionally started with the oldest train.
151. A new command **Restart most recent Schedule** allows to restart the schedule, that has been most recently executed by a particular train. This command is for example useful to continue a schedule, that must have been prematurely terminated for certain reasons.
152. The aspect of calculated block signals cannot only be selected for each block or route in each particular schedule, but also preselected in each block, route or turnout once for all schedules. Among others this allows to accomplish the “yellow” signal aspect also for trains run by **AutoTrain™**. (see page 69).
153. Speed limits, which depend on calculated block signals, cannot only be preset on the level of blocks for all schedules, but also on the level of routes or turnouts once for all schedules. Among others this allows to accomplish speed limits also for trains run by **AutoTrain™**. Furthermore it is possible to lower the speed limits

preset by blocks, routes or turnouts by individual settings for each block or route in a schedule. (see page 69).

154. A new schedule rule can cause trains to reduce their speed to a preset value, when the calculated distant block signal is red due to an unscheduled stop. This causes trains to reduce their speed already in the block before such unscheduled stop applies and can help to improve traffic flow.
155. Turnout position control (see page 104).
156. Conditions and triggers may now contain additional logical groups, that are true, if at least, at most or exactly a certain preset number of items contained in the group have the required state (see page 101).
157. Combined groups can be used in conditions and triggers to check, whether certain trains are located in certain blocks and/or whether these trains are performing certain schedules. They can also be used to check, whether certain blocks are currently involved in certain schedules (see page 102).
158. **Lock all Blocks** is a new command, that can be used to interrupt the operation of your layout without causing trains to perform an abrupt stop (see page 80).
159. **Lock all Schedules** is a new command, that can be used to terminate the operation of your layout without causing trains to perform an abrupt stop (see page 80).

Timetable / Clock:

160. The time, date and other settings of the clock/timetable can be altered outside of edit mode, too.
161. It is possible now to specify a reset time, which is applied during reset of the complete layout and optionally after begin of each session.
162. The display of the clock can be synchronised with the system clock of the computer.
163. The clock can be automatically started and stopped now with system operations called by buttons, macros or even indicators.
164. In edit mode it is possible to limit the display of the timetable to those entries, that are executed at the currently selected date. If this is done, then it is additionally possible to expand the display to show the same content as outside edit mode; i.e. to display the complete operational timetable for the currently selected date.

Turntable:

165. With a new turntable symbol each turntable or transfer table can be operated and controlled via switchboard windows, too.
166. The automatic calculation of the block diagram covers turntable symbols in switchboard windows, too. Routes across and involving turntables or transfer tables are automatically calculated, too. No specific programming or data input necessary to enable a turntable for AutoTrain or automatic train operation.

167. The turn boundary, a difficult to understand feature of previous versions, has been superseded by the following, much more plain approach: each turntable track can be optionally marked as forward or backward. This causes the specified locomotives to leave the bridge via the according tracks in forward or backward direction (see page 123).
168. In addition to the above it is possible to override the direction, in which a locomotive leaves the turntable, on an individual per schedule base (see page 123).

Traffic Control:

169. It is possible now to open more than one traffic control windows at the same time.
170. The traffic control can be pinned to a certain train, to a certain block or to a certain window. This allows several useful applications. See page 84.

Message Window:

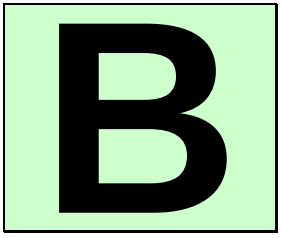
171. It is possible to suppress repeated display of undesired Dr.Railroad messages.
172. The content of the message window can be sorted by column. This is in particular useful to sort the logged messages according to the trains, which these messages belong to.

Hardware and Digital Systems:

173. Selectrix systems and derivatives: push button and on/off switch symbols can be arranged to manipulate several bits of the same Selectrix address simultaneously. This function is useful to operate specific Selectrix compatible decoders, that require manipulation of more than one bit of the same address in one step.

Part II

Fundamentals



1 Introduction

1.2 Variants of Train Control

Train control, i.e. running of model trains on a model railroad layout, is the key aspect of model railroading and hence also for **TrainController™ 7**.

TrainController™ 7 provides a wide range of possibilities to run your trains – from completely manual to completely automatic with a wide range of variants.

The following list provides a brief overview of the different methods to run your trains with **TrainController™ 7**:

(1) Run trains manually, semi-automatically or automatically under full protection, blocking and routing of **TrainController™ 7** along paths and routes, which are automatically activated by the train itself or manually by the end user during the train run. Trains are started ad-hoc, i.e. without specifying destination positions or complete paths in advance (**Run with Interlocking**).
(2) Run trains manually, semi-automatically or automatically under full protection, blocking and routing of **TrainController™ 7** by specifying the start and destination positions at any time during operation by dragging a train symbol with the mouse from its current position to the desired destination position (**AutoTrain™ by Drag & Drop**).
(3) Run trains as before, but specify more than one start and destination position as well as other options such as scheduled wait times, speed limits etc. at any time during operation just before starting the train (**AutoTrain™ Symbol Bar**).
(4) Run trains manually, semi-automatically or automatically under full protection, blocking and routing of **TrainController™ 7** according to schedules, i.e. sets of options, which specify several start and destination positions as well as other options such as scheduled wait times, speed limits etc. and which are created prior to the operating session, i.e. during configuration of the layout. Schedules, which are started manually, by pressing a button, by start- and destination keys, as part of a sequential chain, automatically triggered or by timetables (**Schedules**).
(5) Run trains manually without any protection, blocking and routing performed by **TrainController™ 7** (**Manual Train Control**).

Run with Interlocking

This is the most handy method to run your trains under full protection and routing of **TrainController™ 7**. Just put a locomotive on the track and call the menu command **Run with Interlocking**. The train will immediately start to move, provided that the route ahead is clear. It will then select an appropriate path and continue to travel, until it reaches a dead end or until the path ahead is blocked for another reason. At a dead end it will reverse automatically, if desired, and continue to travel to the opposite direction.

Routes can be treated in different manners for running with interlocking. It is either possible to allow the computer to select and activate all routes requested by the train automatically. It is also possible to leave this to the human operator. In this case the train is stopped in blocks with at least one outgoing route, until one of these outgoing routes is selected and activated by the human operator.

Pros:

- Well suited to accomplish hands-on activity on your model railroad layout including protection, routing and signaling with minimum efforts.
- Easiest way to run trains under full protection and routing.
- Can be spontaneously executed at any time during operation.
- Fastest way to start a train with a **+SmartHand™** handheld under protection of the software.

Cons:

- In general human intervention or specific measures are required to prevent trains from running into tracks, where they must not go.
- Not suited for full automatic operation of the layout, without further measures, because in general human intervention is required to start the train.

AutoTrain™ by Drag & Drop

This is another very convenient method to run trains under full protection and routing of **TrainController™ 7**. Just put a locomotive on the track and drag the symbol of the train on the computer screen with the mouse from its current position to the desired destination position. The train will immediately start to move, provided that the route ahead is clear. It will then select an appropriate path to the specified destination block and travel there, if possible. After arrival at the destination block the train is stopped.

Pros:

- Well suited to move a train spontaneously to a certain location of the layout under full control of the software, protection, routing and signaling, with minimum efforts.

- Very easy way to run trains under full protection and routing.
- Can be spontaneously executed at any time during operation.
- Full control of the destination block, where the train shall go.

Cons:
- Care has to be taken, that there is a possible path between the current position of the train and the desired destination block.
- Not suited for full automatic operation of the layout, because human intervention is required to start the train.

AutoTrain™ Symbol Bar

This is an extension of **AutoTrain™ Drag & Drop**. Instead of dragging a train symbol from its current position to the desired destination the path of the train and other options are specified via the **AutoTrain™ Symbol Bar**. This symbol bar provides more options, that the more simple drag & drop method. The full functionality for automatic running of a train is available here, among others it is possible to specify more than one start and destination block, to enforce inclusion or exclusion of certain blocks, to specify scheduled wait times during the travel, to specify operations, that shall be executed during the travel, to determine, whether the train shall be controlled manually, automatically or by a mixture of both, and so on. The **AutoTrain™ Symbol Bar** is also useful to predefine train runs for automatic operation of the layout.

Pros:
- Well suited to move a train spontaneously to a certain location of the layout with the possibility to apply the full range of options available for train control.
- Well suited, too, to predefine train runs for full automatic operation with minimum efforts.
- Provides the full range of options available for trains running under protection and routing.
- Can be spontaneously executed at any time during operation.
- Full control of the path taken by the train.

Cons:
- Care has to be taken, that there is a possible path between the specified start and destination blocks.
- Not suited for full automatic operation of the layout without further measures, because human intervention is required to start the train.

Schedule

Schedules provide the possibility to predefine train runs in advance and in particular for full automatic operation. Unlike the other methods schedules do not require manual intervention to be started. The full functionality for automatic running of trains is available for schedules, too, among others it is possible to specify more than one start and destination block, to predetermine the exact paths, to specify scheduled wait times during the travel, to specify operations, that shall be executed during the travel, to determine, whether the train shall be controlled manually, automatically or by a mixture of both, and so on.

Pros:

- Well suited for full automatic operation of trains without human intervention.
- Provides the full range of options available for trains running under protection and routing.
- Can be started automatically without human intervention.
- Full control of the path taken by the train.

Cons:

- Require predefinition prior to operation of the layout.

Manual Train Control

Manual train control is performed by putting a train on the track and by driving it with the throttle of the digital system, with the on-screen throttle of **TrainController™ 7** or with a **+SmartHand** handheld without taking any further measures. Although the position of the train can be tracked by the computer, the computer does not activate routes ahead of the train or take corrective action like stopping the train at a red signal. The human operator is completely responsible for routing and stopping. A train driven in this way, however, is protected against other trains running under control of the computer, while other trains are not automatically protected against this train, i.e. the human operator is responsible, that the train operated by him in this way does not run into other trains.

Pros:

- Well suited for manual test runs and basic operation without protection, routing or signaling.
- Can be spontaneously executed at any time during operation.

Cons:

- Low security.
- No automatic routing or signaling.
- Manual control of trains only.
- Number of trains simultaneously operated this way is limited by the skills of the human operator to control and to watch several trains at the same time (usually 1 to 3 per operator).

Comparison Chart

The following chart compares the possibilities of the particular methods and their suitability for certain purposes:

Feature	(1) Run with Interlocking	(2) AutoTrain™ by Drag & Drop	(3) AutoTrain™ Symbol Bar	(4) Schedules	(5) Manual Operation
Block Securing	Yes	Yes	Yes	Yes	No
Automatic Routing	Optional	Yes	Yes	Yes	No
Automatic Signaling	Yes	Yes	Yes	Yes	No
Train Guidance System	Yes	Yes	Yes	Yes	No
Modification by Rules	Yes	Yes	Yes	Yes	No
Automatic Consideration of Speed Limitations	Yes	Yes	Yes	Yes	No
Full functionality for automatic train operation available (e.g. Scheduled Stopovers,…)	No	No	Yes	Yes	No
Can be started with Start and Destination Keys	No	Yes	No	Yes	No
No. of possible Start Blocks per Run	1	1	>=1	>=1	-
No. of possible Destination Blocks per Run	>=1	1	>=1	>=1	-
Start without prior specification of destination Blocks	Yes	No	No	No	Yes
Preset for destination possible	Indirect	Yes	Yes	Yes	Yes
Spontaneous execution w/o prior predefinition	Yes	Yes	Yes	No	Yes
Manual Train Control possible	Yes	Yes	Yes	Yes	Yes
Transfer of Control between operator and computer according to curr. signal status poss.	Yes	Yes	Yes	Yes	No
Automatic Train Control poss.	Yes	Yes	Yes	Yes	No
Effort for Setup/Start	Minimal	Minimal	Medium	Medium	Minimal
Automatic Layout Operation without human intervention	No	No	No	Yes	No
Timetable based Operation	No	No	No	Yes	No

All the methods listed above can be used simultaneously and freely combined. The following modes to run trains manually, namely:

- Run trains manually with the throttle of your digital system.
- Run trains manually with the virtual on-screen throttle of **TrainController™ 7**.
- Run trains manually and fully protected with the physical throttles of the **+SmartHand** handheld control system

can be applied to any manually or semi-automatic operated train for any of the methods listed before. It is also possible to pass each train from manual operation to any of the automatic modes listed before and back or between the particular modes listed above at any time during operation. In short terms: there are almost no limitations.

1.3 Fundamentals of Use

B

User Interface Design

TrainController™ has always been highly customizable, with regard to functionality as well as with regard to usability. **TrainController™ 7** continues this policy consequently. The user interface of **7**, for example, can be very extensively customized to personal needs and taste.

This begins with the overall layout of the **7** user interface. The user interface can be displayed by applying different visual styles. Among others the following styles are available:

- Several Office 2007 styles
- Visual Studio 2008 and 2005
- Native XP
- Office 2003
- Classic Office 2000

Feel free to select the style, that fits best your personal taste.

Window Handling

The particular functions of **TrainController™ 7** are represented in different windows. The general visual design of all windows has been harmonized compared to earlier software versions. The handling of all windows is consistent now, too. For example there is no “main switchboard” anymore with a specific meaning. There are also no windows with fixed size (as the train windows in previous versions) vs. windows with

variable size. In **TrainController™ 7** the size of all windows is variable and can be adjusted to personal taste.

Each window can appear in one of the following states:

- Docked to one of the borders of the main window.
- Docked to another window.
- Floating at any location on the computer screen; individually or grouped/docked together with other windows.
- Tabbed with other windows – as one of several tabbed documents in the background of the main window or together with other windows in a floating or docked frame.
- Auto-Hidden while not active with quick access via a button on any side of the main window.

In previous versions of **TrainController™** each window could be docked to the main window or individually floating, but not more.

The possibility to group related windows together in **TrainController™ 7**, either docked or tabbed, in the main window or in a floating frame somewhere on the computer screen, opens interesting possibilities. It is now possible, for example, to arrange a set of related windows for control of one part of the layout together in one group and to arrange another set of related windows for another part of the layout in another group. Such group of related windows can be then moved, resized, hidden, restored and even docked and tabbed together with other groups of windows just like one single window, which makes it very convenient and effective to manage sets of windows, that belong together. An example of such related windows is a switchboard combined with a dispatcher window, that shows just the block diagram of this switchboard (**TrainController™ 7 Gold** allows to create multiple block diagrams for multiple switchboards now and to open more than one dispatcher window, though more about this later). No matter how many windows you need to open to represent your entire layout, you will find a window arrangement, that fits your needs and personal taste optimally.

TrainController™ 7 makes docking of windows much more intuitive and easy, too, by showing docking markers for each window, that is currently being dragged over the computer screen. The docking markers intuitively indicate, where to move the mouse in order to dock the dragged window to the desired location. By moving the mouse to a docking marker an additional docking outline provides a clear preview of the docking effect. Thanks to this feature, which has been borrowed from state-of–the-art professional software development systems there is no more puzzling, where a window will be finally docked.

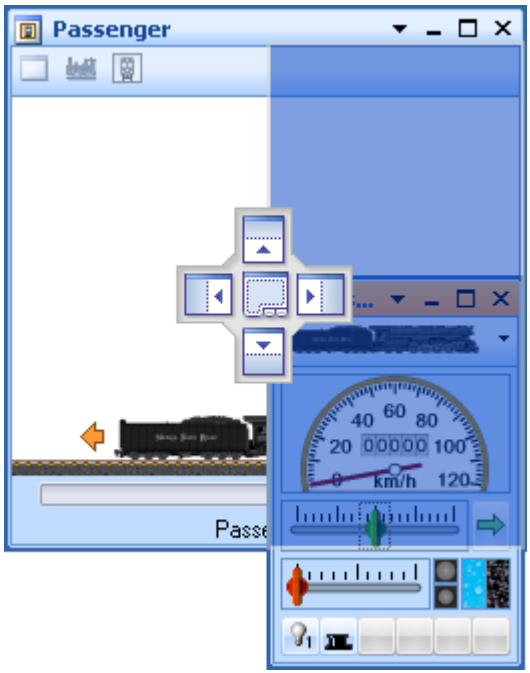


Diagram 2: Docking a Train Window to the right of a Traffic Control

TrainController™ 7 now stores the window arrangement individually for each project, while previous versions applied a common arrangement to all projects. Especially those users, who work with different data files for different projects will appreciate the possibility to create and to save an individual arrangement of windows for each particular project.

Even for the rare case, that the windows on your computer screen are misaligned and you don't know how to return to a consistent state, provisions have been made: with a specific menu command it is possible to load the current data file once more with a default window status, which can be used again as a starting point for an individual arrangement.

Window Customization

The layout of those windows, which contain the most comprehensive data or which are used most frequently, the switchboard, dispatcher, train and explorer window, can be highly customized to personal needs and taste.

You can play with all settings without any risk, because in **TrainController™ 7** all customizable windows provide the possibility to reset the settings to factory defaults.

Customization of Menus, Tool Bars and Keyboard Accelerators

It is also possible to customize the content of menus and tool bars and to change keyboard accelerators.

New menus and tool bars can be created, commands can be added or removed from menus and tool bars and existing commands can be changed. It is possible to create new menu and tool bar symbols for commands, that do not have a symbol associated with it by default, or to change existing icons with a built-in icon editor.

It is furthermore possible to display all menu and toolbar icons in large size.

Keyboard accelerators can be changed. It is also possible to assign keyboard accelerators to commands, that do not have a keyboard shortcut associated with it by default.

2 Controlling Turnouts and Signals - The Switchboard

2.2 Size and Appearance

Additionally to the general customization features of previous versions **TrainController™ 7** now provides the following additional customization features:

- A new option allows reset of all display options to factory defaults.
- Six highlighting styles can be applied generally or differing for plain tracks vs. turnouts, occupied vs. non-occupied track symbols as well as track symbols in active routes. The possibilities to apply different styles and colours to different highlighting aspects provide virtually countless layout combinations and allow reproduction of almost each highlighting style of prototypical plug panels.
- The highligthing of occupied track elements can be turned off globally; or it can follow the activation of routes, i.e. only those track elements are highlighted, which are contained in a currently active route; or the highlighting can be controlled by individual assignments to indicators as in previous versions. The highlight color of occupied track sections can be controlled by the reserving train, if any, as in previous versions, or by the color of the occupied indicator or by specifying a constant color value.
- Routes, that are automatically created by the automatic creation of the block diagram, can be optionally completely highlighted now. This means, that all track elements located on the track connection between two blocks are highlighted by an active route, even if these track elements are not actually contained in the route. In previous versions highlighting was limited to those track elements, that are actually contained in the active route. This display mode is still available as an option, too.
- Active routes can be highlighted with individually specified colors (as in previous versions), or with the color of the reserving train, if any, or with a color, that is common for all active routes.
- It is optionally possible now, to display the name of the associated block in the block diagram, too, when edit mode is turned off. Previous versions of the software displayed block names solely in edit mode.
- Switchboard symbols can be displayed in five different sizes ranging from 12x12 to 28x28 pixels per symbol / switchboard cell.
- Images are automatically scaled, if the symbol/cell size changes.
- In **TrainController™ 7 Gold** it is possible to create custom switchboard symbols for signals, push buttons, on/off switches, toggle switches and routes with an integrated bitmap editor and to assign such custom symbols individually to each accord-

ing switchboard object. Custom switchboard symbols can be transferred between different data files by export and import.

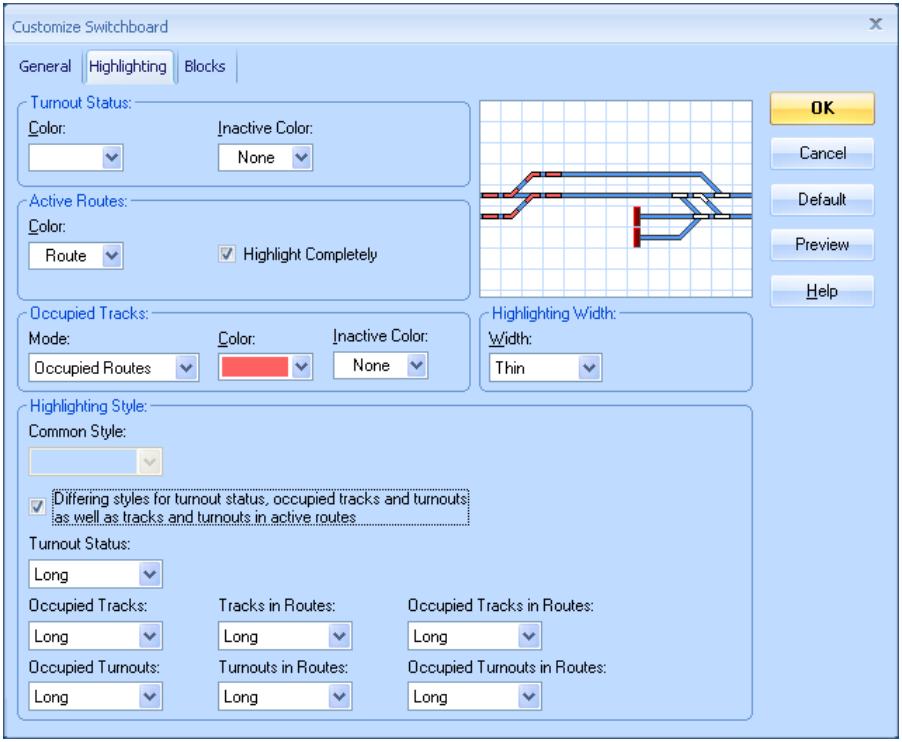


Diagram 3: Customization of Highlighting in the Switchboard

2.3 Drawing the Track Diagram

B

In addition to the track elements provided in other versions some new track elements are available in **TrainController™ 7 Gold**:

- Space-saving turnouts as *left-* and *right-hand turnouts*, *wye turnoust* and *three-way turnouts*
- *Connecting tracks* for space-saving turnouts
- *Left*, *right* and *symmetric crossings* for use with space-saving turnouts and their connecting tracks.

Space-saving turnouts need less space in the switchboard in certain situations, e.g. in case of crossovers. Furthermore the look of certain prototypical switchboard panels, that use such symbols, too, can now be replicated more realistically.

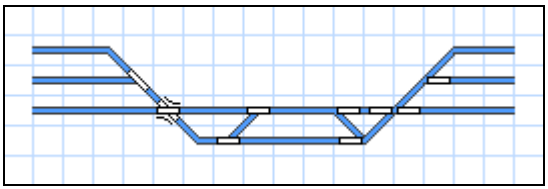


Diagram 4: Normal and Space saving Turnouts

Diagram 4 shows two identical situations, on the left side drawn with normal turnouts and on the right side drawn with space saving turnouts. The left part requires more space, even though a double slip switch can be used there.

This diagram illustrates also, that normal turnouts and space saving turnouts can be combined in the same track diagram without any problems.

TrainController™ 7 Gold leaves the choice to you, which type of turnout symbols you use, the normal, the space saving or both. This enables you to create a track diagram layout, that fits best to the available screen space, to the prototypical panel style, that you may want to replicate, or just to your personal taste.

2.8 Self-provided Switchboard Symbols and Images

Self-provided Symbols

In **TrainController™ 7 Gold** it is possible to create switchboard symbols with an integrated symbol editor yourself and to display them in the switchboard. Such symbol must only be drawn once and can be displayed many times in the switchboard.

Switchboard symbols are inoperable and mainly used for small switchboard graphics and icons, which extend the default stock of predefined switchboard symbols.

Symbols can be arranged in the background, i.e. behind the track diagram, or in the foreground of the switchboard. Symbols in the background can be covered by track elements. Symbols in the foreground can cover track elements.

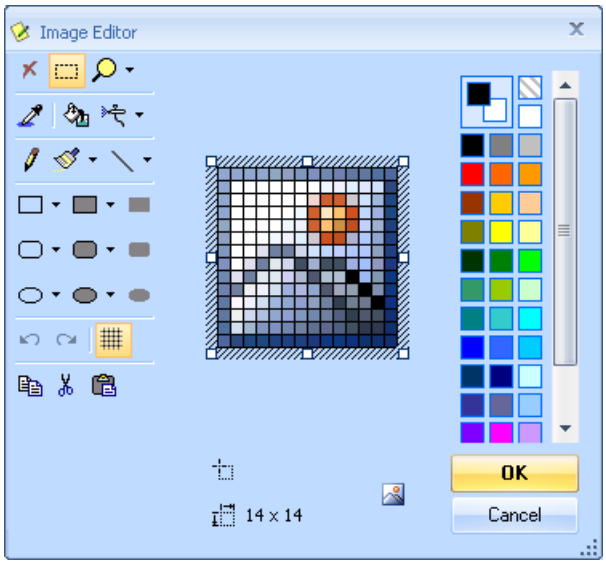


Diagram 5: Creation of self-provided switchboard symbols with the built-in image editor

Self-provided symbols can be transferred between different data files by export and import.

Images

It is possible to display images stored in external bitmap, gif or jpeg files in your switchboard. The following possibilities are provided:

Images can be arranged in the background, i.e. behind the track diagram, or in the foreground of the switchboard. Images in the background can be covered by track elements or by images laying in the foreground. Such images can be used to display landscape structures like meadows or rivers. Images in the foreground can cover track elements and can be used to display buildings, bridges or tunnels.

It is additionally possible to fade out portions of an image, i.e. to draw portions “transparently”. This is useful if images with irregular shapes are drawn. This is done by drawing the parts of the image, which shall be drawn transparently, with a certain color, which is not used elsewhere in the image.

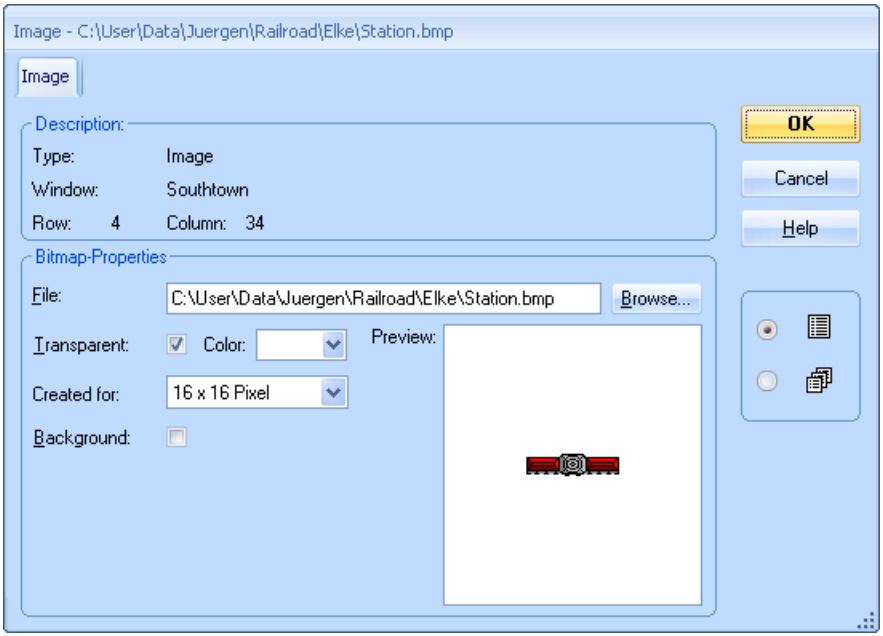


Diagram 6: Arranging an image

3 Train Control

3.1 Introduction

The Train Window

B

Train windows can be used to operate trains manually with the mouse or keyboard of the computer or to watch the status of running trains during operation.

Train windows contain various controls and instruments, that are used to operate or to monitor each train.

A sample *Train Window* is displayed below:

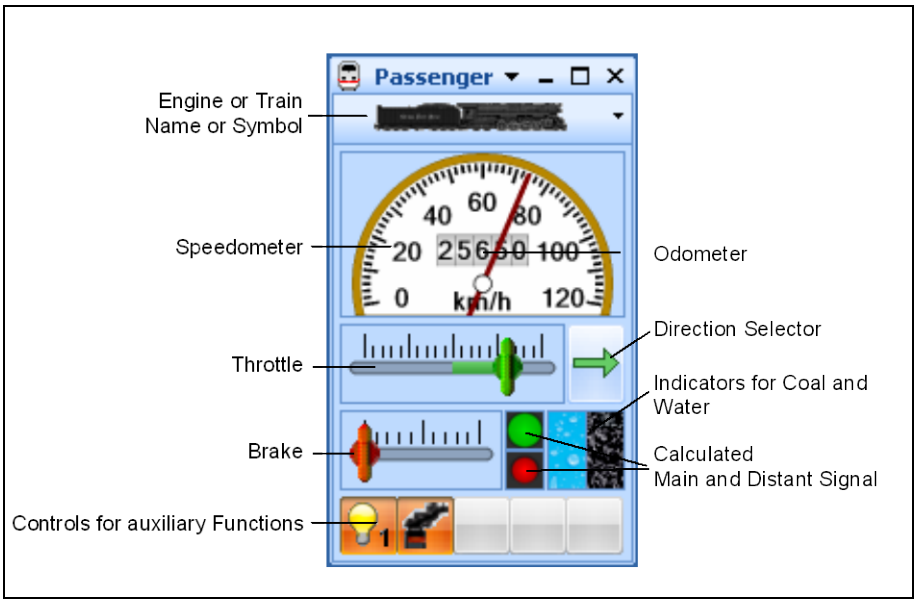


Diagram 7: Train Window

The design of the *train window* provided by **TrainController™ 7** has been improved in many aspects compared to prior versions of **TrainController™**.

- The basic design of all controls and instruments of the train window has been revised to fit to the new design of the new user interface.
- The size of the train window can be continuously adjusted now. This is possible on the fly, like any other window, by dragging the borders of the train window with the mouse.
- It is also possible to define an *ideal size* for train windows. The ideal size of each train window can be restored at any time with a single mouse click.
- The symbol of the currently selected train is displayed in the title bar now.
- The sizes of the particular groups of instruments can be individually adjusted now. In previous versions of **TrainController™** it was only possible to set all instruments collectively to the small or to the large display mode. Furthermore **7** is able to switch dynamically between the small and large display mode according to the available space. Finally it is possible to hide individual and not needed groups of instruments in order to save space on the computer screen.
- The number of steps for operation of the throttle and brake control with the computer keyboard can be individually set now. A specific setting allows the throttle to follow the number of physical speed steps of the controlled decoder.
- The throttle can be arranged now with the zero position in the middle or on the left.
- The throttle can be set to operate **train oriented** or **layout oriented**. A **train oriented** throttle causes the train always to move forward, when the direction control is pointing to the right, and to move backward, when the direction control is pointing to the left. A **layout oriented** throttle causes the operated train to move to the left on the layout, when the direction control is pointing to the left, and to move to the right on the layout, when the direction control is pointing to the right. This setting is only effective for trains currently assigned to a block. If the block is aligned vertically on the computer screen, then the train will move to the top or bottom, when the direction control is pointing to the top or bottom, respectively. This setting emulates the characteristics of a former throttle for analogue DC railroads.
- The throttle can be set to operate the speed of the train (less realistic, but more convenient) or the power of the train (more realistic, but less convenient). If the throttle controls the speed, then the train is always accelerated with the maximum engine power. Dragging the throttle to a certain position causes the same effect, as if the throttle were first dragged to the maximum position and then reduced to this position, when the train reaches the corresponding speed. If the throttle controls the power, then the train is always accelerated with the power, that corresponds to the slider position. This provides more realistic train control, because many throttles of real railroads actually control the effective power rather than the speed of the train. In such cases the speed indirectly “follows” the effective power. This option,

however, also requires more complex user intervention for train control and is less convenient than direct speed control. It is also possible to set individuals throttles to operate trains without any momentum.

- The colors and layout of the speedometer and odometer can be very individually customized in **7**, too, with a variety of options.
- All the above settings can be reset to factory defaults at any time, if desired.

3.2 Engines

B

An *engine* describes different properties of one of your model engines. These are prototypical attributes like maximum speed or power, or model related properties like digital address or auxiliary functions.

For operation of your engines it is sufficient to enter each engine with its *digital address* in **TrainController™** To specify the digital address or other attributes mark the appropriate engine in the Train List or in a Train Window and select the **Properties** command of the **Edit** menu. Once an engine is entered with its digital address it is then possible to control it with the Train Window.

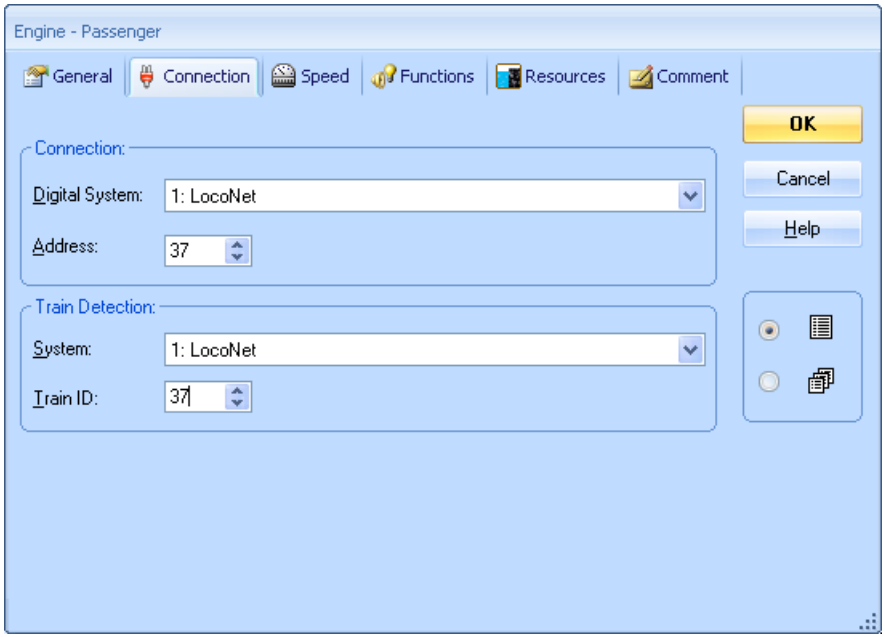


Diagram 8: Digital Address of an Engine

3.3 Throttle and Brake

B

The *throttle* is used to control the speed of each *engine*. The zero position of the throttle is located in the middle by default. When the slider of the throttle is in the rightmost position, the engine is running forward with maximum speed. Conversely the maximum backward speed is achieved by pulling the slider to the leftmost position.

It is also possible to set the zero position of the throttle to the leftmost position of the throttle control. In this mode the maximum forward or backward speed is achieved by pulling the slider to the rightmost position. The direction of the engine is controlled by the separate direction selector.

The above sections describe the **train oriented** mode of the throttle control. In this mode the throttle causes the train always to move forward, when the direction control is pointing to the right, and to move backward, when the direction control is pointing to the left. In the **layout oriented** mode the throttle causes the operated train to move to the left on the layout, when the direction control is pointing to the left, and to move to the right on the layout, when the direction control is pointing to the right. This setting is only effective for trains currently assigned to a block (see section 5.2, “Blocks”). If the block is aligned vertically on the computer screen, then the train will move to the top or bottom, when the direction control is pointing to the top or bottom, respectively. This setting emulates the characteristics of a former throttle for analogue DC railroads.

An additional instrument to control the speed of a train is the *brake.* Pulling the slider of the brake decelerates the train. The brake is an additional aid. For simplicity it is possible to control the speed with the *throttle* only foregoing the *brake.*

For each *engine* you can specify the *maximum scale speed.* This value is used as the maximum speed with which an engine is controlled by **TrainController™** To run an engine with maximum speed the throttle slider must be pulled to the maximum position.

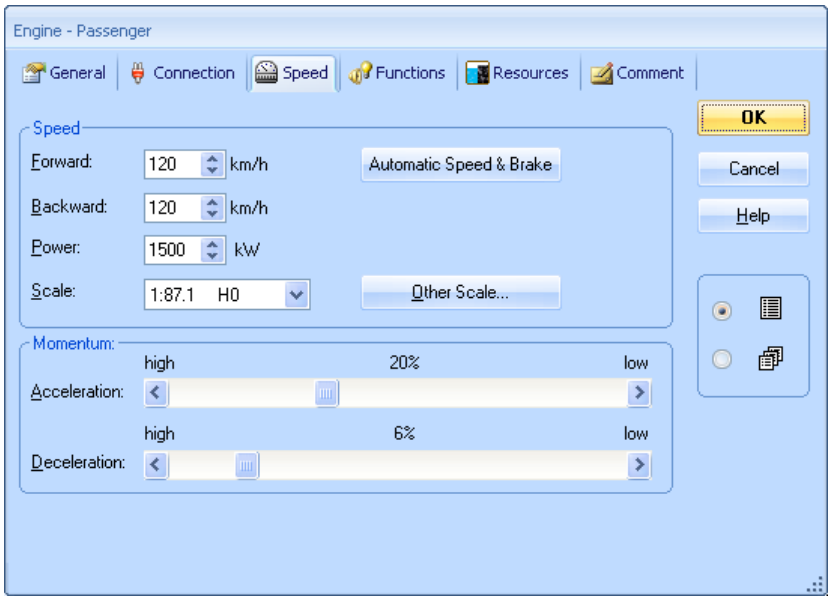


Diagram 9: Speed Properties of an Engine

For each *engine* you can also specify the *threshold speed*. This is the minimum speed at which the engine runs smoothly. The threshold speed is used if the throttle slider is moved out of the zero position. In this way "dead zones" near the zero position of the slider are avoided. For engines which will run automatically under control of the *Dispatcher* (see chapter 4, "The Visual Dispatcher") it is recommended to adjust the threshold speed accordingly.

3.5 The Speed Profile

Trimming the Brake Compensation

In addition to the five procedures to perform the measurement of the speed profile **TrainController™ 7 Gold** provides two further procedures, that support the trimming of the brake compensation.

The brake compensation is a value, that describes the braking behaviour and deceleration momentum of the physical engine. This value is used to compensate additional deceleration delays - e.g. caused by the engine decoder or by a flywheel mass - when an

engine is decelerated. If this engine tends to exceed pre-set braking ramps or stop distances when it is slowed down, then increase this value. The default value is 0, which means, that no compensation is performed. Please note: this option has only an effect in conjunction with combined brake/stop indicators or Virtual Contacts and only when engines are decelerated before reaching their location.

The brake compensation cannot be actually measured. The optimal value must be found by trial and error. Nevertheless **TrainController™ 7 Gold** can help you to find the optimal value most efficiently. An initial estimated value is determined with the last two passes of the automatic measurement of the complete speed profile as outlined in another section. This initial value can be optimized with one of the procedures listed below:

Verifying the brake compensation by decelerating an engine from a pre-set speed to zero. The deceleration starts, when a certain momentary contact is triggered.

Verifying the brake compensation by decelerating an engine from a pre-set speed to zero. The deceleration starts, when a certain occupancy sensor is triggered.

The procedure can be performed with the same indicators and the same track sections, that have been used for the measurement of the speed profile. In case of momentary contacts, the indicator can be used, that marks the start of the measuring track (there called “Start”). In case of occupancy sensors, the indicator can be used, that marks the measuring track itself (there called “Centre”).

To start the test run put the engine on the track in a certain distance left of the selected indicator (“Start” or “Centre”, respectively) and select a typical speed, with which the affected engine enters those blocks, where it usually has to stop. Half of the maximum speed is often a good choice. In the field **Length** of the dialog box specify the length of an estimated average braking ramp used for your blocks.

Then press **Start**, to start the test run. **TrainController™ 7** now accelerates the engine to the specified speed and when the selected indicator is turned on, it tries to decelerate and stop the train at a location, which corresponds to the value specified as **Length**. After the engine stopped, measure the distance between the point, where the engine is located now and the point, where the indicator was turned on. If this distance corresponds to the value specified as **Length**, you are ready. The brake compensation fits.

If the actual distance is higher than the specified **Length**, then increase the brake compensation and repeat the test run. If the actual distance is lower, then decrease the brake compensation and repeat the test run, too.

Repeat the test run, until the brake compensation fits. After this has been done repeat the complete procedure to trim the brake compensation for the backward direction of travel, too.

It is important to perform the complete measurement of the speed profile prior to trimming of the brake compensation.

The contact spot of the engine should be determined and specified, too, prior to trimming of the brake compensation.

3.6 Headlights, Steam and Whistle

Operation of Function Only Decoders

TrainController™ 7 Gold supports the operation of train functions controlled by additional function-only decoders. This is accomplished by specifying an alternative address for each auxiliary function controlled by such decoder.

If a particular locomotive, for example, is equipped with a regular engine decoder with digital address 3 and an additional function decoder with digital address 27, then specify 3 as regular digital address for this engine and set the alternative address to 27 for each function controlled by the function decoder.

The number of alternative addresses, that can be specified for each vehicle, is not limited. Each engine function can carry its own individual alternative address. In other words: **TrainController™ 7 Gold** can operate all decoder functions, regardless, whether they are installed in the main engine decoder or in an additional function decoder, and regardless, how many additional function decoders are installed in a vehicle.

5 The Visual Dispatcher

5.2 Blocks

Block Diagrams

Like a human operator must know the overall structure of the model railroad layout the *visual dispatcher* needs to know this, too. This structure is represented by one or more diagrams, that contain blocks and, routes between blocks. These diagrams also display the various itineraries of your trains. Such diagrams are called *block diagrams* of the layout. They describe the track and block layout of your entire model railroad in rough outline.

Block diagrams are displayed in separate windows, the *dispatcher windows*.

Normally each switchboard, that you create for your layout and that contains blocks, corresponds to a block diagram. These block diagrams are created automatically by **TrainController™ 7** by using the track layout drawn in the switchboard and the information about the blocks contained therein. To enable **TrainController™ 7** to create (“calculate”) a block diagram for a switchboard, it is necessary to specify the positions of the blocks in the track diagram of the switchboard, if there are any. This is done with the help of *block symbols*.

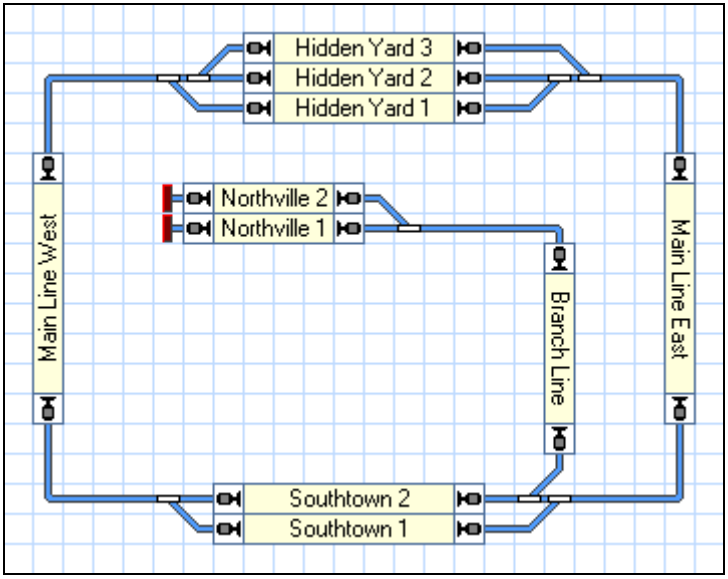


Diagram 10: Switchboard with Blocks

By creating a switchboard, drawing a track diagram in it and inserting block symbols at positions, where blocks are located, **TrainController™ 7** will be caused to automatically calculate a block diagram for this switchboard. All connecting routes will be automatically calculated, too. No extra human intervention is necessary to accomplish this and all routes with all contained turnouts are created automatically, too.

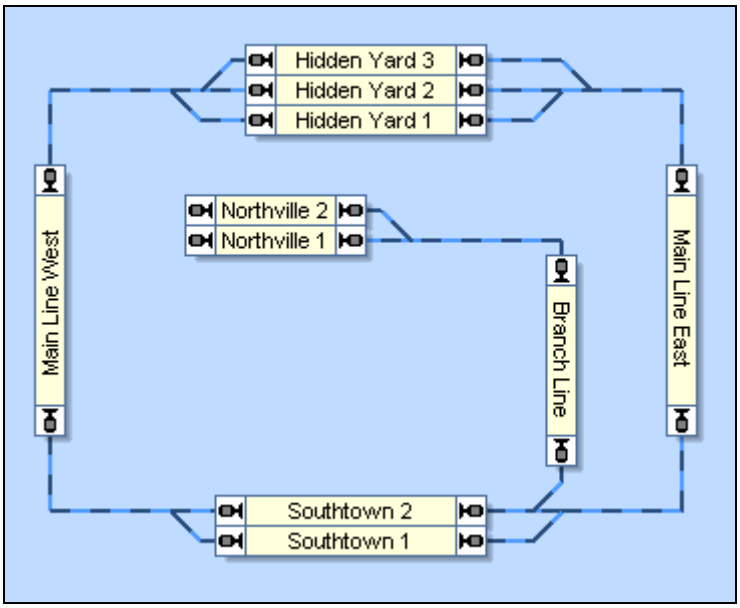


Diagram 11: Main Block Diagram in the Visual Dispatcher

Blocks are displayed on the computer screen by rectangular boxes. The blocks are connected to each other by routes, which touch each box graphically at a smaller side. These routes are drawn as lines.

Please note that the block diagram represents the track layout in rough outline. The actual track connection between “Main Line West” and “Hidden Yard 3”, for example, contains two switches. These switches are not drawn in the block diagram in detail or as separate objects. Instead a line between both blocks is created, that indicates, that there is a track connection between both blocks.

In order to enable **TrainController™** to calculate the block diagram automatically note the following:

- Draw the complete track diagram of your layout with all turnouts and crossings and without any gaps in a switchboard window.
- Create block symbols for all blocks of the layout, place them according to their location on the actual layout and ensure, that they are turned horizontally or vertically according to the track symbols, to which they are attached.
- Ensure that the blocks are connected to each other by track symbols without any gaps. The connecting tracks must touch the blocks at the smaller sides.

For specific purposes it is also possible in **TrainController™ 7 Gold** to place blocks on diagonal track symbols. For technical reasons the size of such blocks is automatically shrunken to one single switchboard cell. Adjacent track elements, which shall connect such *mini block symbols* with adjacent blocks must touch the block accordingly in appropriate opposite corners.

When working with **TrainController™ 7** you may encounter, that switchboards and their corresponding block diagrams seem to look almost identical at first glance. But this is not actually the case. Switchboards contain the details of the track diagram, i.e. each particular track symbol and turnout and also additional objects like signals, push buttons etc. Switchboards are also the base to operate your layout, i.e. to perform manual interventions during operation. By contrast block diagrams display routes between blocks rather than single track or turnout symbols and no additional objects like signals or buttons. Block diagrams mainly serve to manage the blocks and routes and to define and manage predefined itineraries for your trains ("schedules"). They can also be used to monitor the traffic on your layout but are usually not used for manual intervention. In many cases you will display the block diagrams only during edit mode to manipulate your data but hide them during operation.

TrainController™ 7 Silver is limited to one block diagram in total, even if more than one switchboard exists.

TrainController™ 7 Gold allows to work with as many switchboards and block diagrams as necessary to represent your complete layout.

Even though block diagrams are normally automatically created by the software, it may be necessary under certain circumstances, however, to integrate a part of your layout into the block system of the dispatcher, which cannot be represented in a switchboard window. For this purpose **TrainController™ 7** allows to create additional, manually drawn ("custom") diagrams, too.

Routes between Blocks

B

In order to let trains run from one block to another the blocks must be linked together. This is done with the help of *routes*. In the block diagrams routes are represented by lines that connect one block with an adjacent block.

Each block has two entries/exits. If a block is passed horizontally, then the entries/exits are graphically located on the left and on the right side of the block. If a block is passed vertically, then the entries/exits are located at the top and at the bottom. Each route begins at the entry/exit of a block and ends at the exit/entry of an adjacent block.

The following image explains the terms once more:

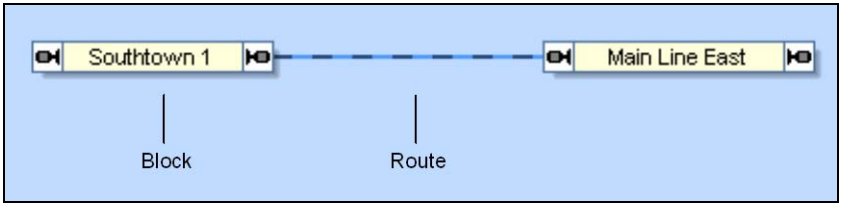


Diagram 12: Blocks and Route

In the diagram displayed above the blocks “Southtown 1” and “Main Line East” are connected with a route.

Then the necessary routes are automatically created according to the track connections between the blocks in the track diagram of your switchboard.

Linking Switchboards together - Connector Symbols

If you are working with more than one switchboard and there are track connections between parts of your layout, that are represented by different switchboards, then these track connections can be represented by *connector* symbols.

Connector symbols are inserted into the track diagrams of your switchboards in a similar way like blocks. Each connector symbol has a name of up to 2 letters or digits, which is displayed in the switchboard, too. To link a certain track symbol in one switchboard to a track symbol in another, insert a connector symbol next to each track symbol in both switchboards and assign the same letters or digits to both. Associated connector symbols are namely identified by identical names. Two connector symbols are associated with each other, if they have got the same name. It is not possible to create more than two connector symbols with the same name.

Connector symbols located in a switchboard are also inherited to the associated track diagram. Furthermore **TrainController™ 7 Gold** automatically creates a hidden route between each two associated connector symbols. This route represents the said track connection between the two switchboards. From now on **TrainController™ 7 Gold** is aware, that trains can travel from one switchboard to the other by passing these connectors and the hidden route in between.

If you like you can also use connector symbols to connect one part of your track diagram with another in the same switchboard. In this case **TrainController™ 7 Gold** will also create a hidden route, if both connectors are contained in the same switchboard diagram. This is sometimes useful for large, complex track diagrams, where omitting certain track connections improves the clarity of the display.

It is also possible to insert connector symbols into custom diagrams to connect such diagrams with switchboards or again other custom diagrams.

5.4 States of a Block

Unidirectional Blocks

In **TrainController™ 7 Gold** each block can be specified to be unidirectional. An unidirectional block can only be passed in a certain direction of travel.

Locking the entries of Blocks

Each block can be temporarily locked during operation. Locked blocks cannot be reserved by running trains. A train, that is already located in a block, when the block is locked, might stay there, though, and leave the block later. A lock does not also have an effect for a train, that has already reserved the block, before the lock is set. This train may proceed into the locked block.

Locks are directional. It is possible to set an individual lock for a particular direction of travel. This allows to permit trains to pass the block only in one direction of travel. For this reason these locks are also called entry locks. The lock prevents trains to enter the block via the locked entry, while trains approaching the block from the opposite direction are not affected by this lock.

Please note that locking of a block affects all trains.

In **TrainController™ 7 Gold** locking the entry of a block causes a similar effect like an unidirectional block (see above), i.e. both prevent trains from passing a block in a certain direction of travel. There are some important differences, however, between unidirectional blocks and blocks, which are locked to certain direction of travel:

- An entry lock can be set and removed at any time during operation. Unidirectional blocks can only be changed in edit mode. Hence an entry lock prevents a train from passing the affected block in a certain direction only temporarily while an unidirectional block does this permanently.
- Entry locks are treated as temporary obstacles. It is possible to establish a path for a train run (e.g. via **AutoTrain™**), that passes a block in a (temporarily) locked direction. The train may even approach a locked block, awaiting that this lock is lifted sooner or later.
- Unidirectional blocks are treated as permanent obstacles. It is not possible to establish a path for a train run (e.g. via **AutoTrain™**), that passes a block in a (permanently) disabled direction.

If a certain track section is intended to be always used in a certain direction of travel, then define the according block as unidirectional. If you want to prevent trains from passing a block in a certain direction of travel for a limited space of time only during operation, then use an entry lock. An entry lock can for example be used to lock the opposing entry of a bi-directional single-track section, which is currently occupied by a train, against opposing trains. Using unidirectional blocks for such track section would not be adequate, because this would not allow to operate the track section alternately in both directions.

5.6 Blocks and Indicators

B

For proper operation the *dispatcher* must be able to detect, whether a train occupies a specific section of your railroad or when a train passes a specific point on your railroad. This detection is done with *contact indicators*.

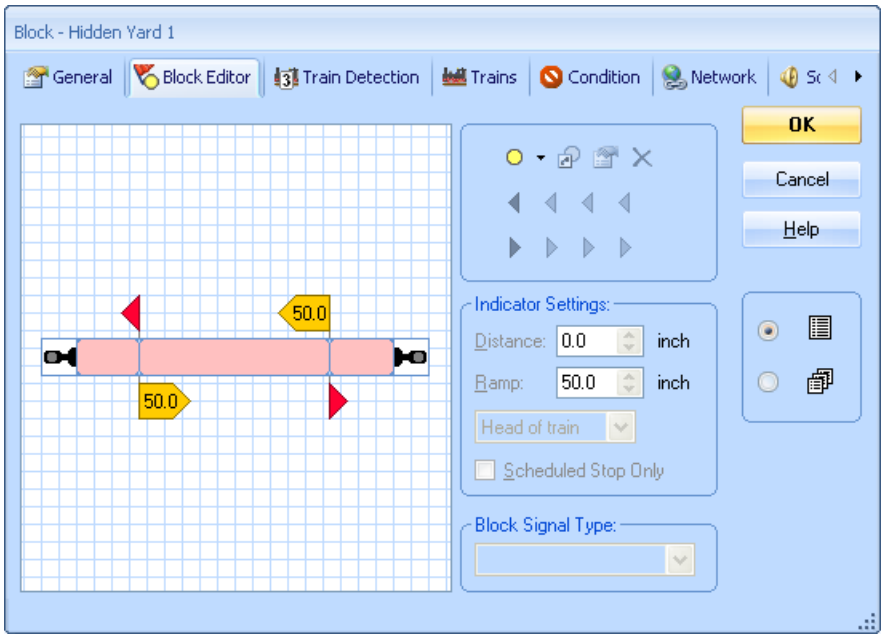


Diagram 13: The Block Editor

In order to establish a block, *contact indicator symbols*, which represent the track sensors located in this block, are created and assigned to the block. This is done with the *block editor*, which is displayed in Diagram 13. If at least one of the indicators contained in a block is turned on, then the block is assumed to be occupied. The actual layout positions of the sensors assigned to the block determine also the location of the block on your railroad.

The block editor shows an edit area with the current configuration of the block. Contact indicators are displayed as red rectangles in the center of the editor. Usually these rectangles represent the occupancy sections associated with each indicator (in case of occupancy sensors) or the point in the block, where the indicator is triggered (in case of momentary contacts like reed turnouts, mechanical contacts, etc.). Each physical sensor located in the block is represented by one indicator rectangle. The location and size of these indicator rectangles can be customized and do not have any impact for the operation of the program, but if properly arranged they can visualize the section, that is covered by a certain contact sensor.

In order to have control over the exact location, where a train shall stop or change its speed inside a block, certain sections can be marked as stop, brake or speed sections (see section 5.7, “Stop, Brake, Speed and Action Markers”) or combinations of these.

To establish a block on your railroad, it is necessary to install the required sensors. Depending on the principle of the used contact sensors it may be necessary to electrically insulate the track section belonging to a certain contact sensor from adjacent sections. Whether electrical insulation is necessary or not depends solely on the contact sensors being used. The software does not require electrical insulation of your blocks.

- The software does not require that a block is electrically insulated from other blocks. The used sensors might require this, though.
- Blocks usually contain several indicators. If these indicators represent isolated or separate track sections then several track sections are contained in the same block (see also 5.8, “Arranging Indicators and Markers in a Block”).
- The same indicator cannot be assigned to several blocks. In particular you should install your sensors on your layout in a way, that each sensor section is associated with at most one block. If you use a train detection system (see 5.5, “”) then each train detection section or zone, respectively, must be associated with at most one block, too.
- Even though it is possible to assign indicator symbols to a block, which are already contained in other windows, this feature is mainly provided for reasons of compatibility to previous software versions or for very specific purposes. Usually you should create each indicator symbol, which is contained in a block, with the block editor displayed in Diagram 13.

5.7 Stop, Brake, Speed and Action Markers

B

A block is established by creating and assigning one or more *indicators* to it. If at least one of these indicators is turned on, then the block is assumed to be *occupied*. The indicators are used for indication of occupancy.

It may happen, that a train has to stop or to change its speed when passing a certain block. This is for example the case, when the block ahead is not available, when the train shall stop inside the block for a certain amount of time or when another speed limit applies in the subsequent block. The exact locations, where trains shall stop or change their speed inside the block are determined by marking certain indicators with *stop*, *brake* or *speed markers*.

Stop and Brake Markers

Let us assume that a train approaches a certain block. That means, that none of the assigned indicators was activated before and that at least one of these indicators is activated now. The block is marked as occupied now and the train continues with unchanged speed. If the train reaches a location in the block, which is marked by a brake marker for the current *direction of travel* (see section 5.3, “”) and the train has to stop inside this block, then the train is decelerated to its *threshold speed*. The braking ramp can be set as desired individually for each brake marker, too. If the train reaches a position, which corresponds to a stop marker for the current *direction of travel* and the train has to stop inside this block, then the train is stopped here.

A stop marker determines a point in a block, where trains stop. Stop markers are represented in **TrainController™ 7** by red arrowheads pointing to the direction of travel, in which they apply. A brake marker determines a point in a block, where trains, that have to stop in a block, begin to slow down. Brake markers are represented in **TrainController™ 7** by yellow arrowheads.

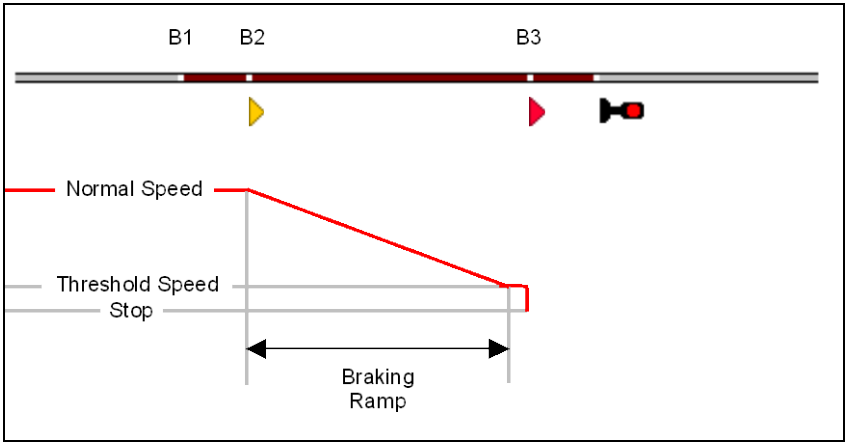


Diagram 14: How Brake and Stop Markers work – Occupancy Sensors

Diagram 14 shows a block, which is equipped with three occupancy sensors. The left entries to the sensed track sections are labelled with B1, B2 and B3.

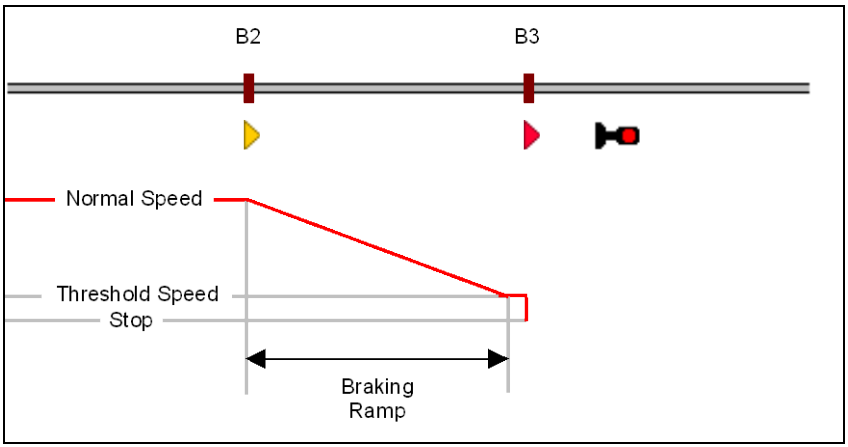


Diagram 15: How Brake and Stop Indicators work – Momentary Track Contacts

An alternative, but for this discussion almost equivalent situation shows Diagram 15. It contains a block, that is equipped with two momentary contacts. These contacts are labelled with B2 and B3, too.

B3 is marked with a stop marker ( ) effective for trains travelling to the right. B2 is marked with a brake marker ( ) effective in the same direction. B1, that applies only to the first diagram, is neither marked as brake nor as stop marker. B1 is used only for occupancy detection.

The red line shows the speed of the train. It is assumed that the train shall stop in this block, i.e. at B3. When the train enters the block at B1 nothing happens, because B1 is only used to report the entry into the block. When the train reaches B2, it is decelerated to its threshold speed. The braking ramp can be specified individually for each brake marker. After deceleration the train proceeds at threshold speed until it reaches B3. When the train reaches B3, it is stopped without delay.

Diagram 13 shows the same situation as Diagram 14 configured in the block editor.

If the train does not have to stop in this block, then it passes all indicators and markers without any speed change.

If the stop marker B3 is missing, then the train will run with normal speed to B2 and stop there without delay. If no stop marker is assigned to a block, then the first appropriate brake marker is used as stop marker. If only the indicator B1 is existing without any markers, then the train will already be stopped at B1. If no marker is assigned to a

block, then the first triggered indicator implicitly defines a stop marker, too. If necessary, a train is stopped in a block anyway, even if no brake and stop markers are assigned.

This examples illustrates also, that proper operation of brake markers requires correct adjustment of threshold speed of each affected train! If this is not the case, the train will be decelerated to an undefined threshold speed. Normally this speed will be too low to run the train properly and the train will stop before reaching the stop marker.

A stop, brake or speed marker is always associated with an indicator. Usually this is a contact indicator, that represents an occupancy section or momentary contact installed on your layout. A stop, brake or speed indicator is valid for a certain direction of travel. The marker usually takes effect, when a train running in this direction enters the associated occupancy section or touches the associated momentary contact. It is also possible to specify a distance for each stop, brake or speed marker. In this case the marker takes effect, when the train has passed this distance after entering the associated occupancy section or after touching the associated momentary contact. Such markers are called *shifted* stop, brake or speed markers.

While each marker is always associated with exactly one indicator, it is possible to mark the same indicator with several markers. The same occupancy section, for example, can be used to slow down passing trains (brake marker) and to stop trains in a certain distance behind the border of the section (shifted stop marker). This is accomplished by adding a brake and a stop marker to the same contact indicator, which represents the occupancy section, and by specifying an appropriate distance for the stop marker. It is even possible to add more than one brake, stop or speed marker to the same indicator or to the same block. The assignment of two stop markers to the same indicator, for example, is useful, if different trains shall stop at different positions (e.g. advance of freight trains to the block signal at the end of the block while passenger trains stop at the middle of the platform). For this purpose the validity of a stop, brake or speed marker can be limited to certain trains.

Please note that a brake marker is only effective if the train has to stop in the same block. As a consequence brake and stop markers that belong together must be contained in the same block.

The same indicator can be marked with stop or brake markers for one or both directions of travel. It is even possible, that a certain indicator is associated with a stop marker in one direction and with a brake marker in the opposite direction.

It is recommended to locate the sensors, which correspond to the stop markers, at positions, which ensure, that even long trains completely fit into the block.

If an engine or train passes a sequence of blocks and a certain block is not available or must be passed at restricted speed, then the train is stopped or decelerated in the previous block. Brake and stop markers control, if a train may exit a certain block. For this reason **TrainController™** always assumes, that stop markers are usually located near the exit of each block with reference to the direction of travel they are effective.

When a train enters a block, the dispatcher checks if there is a route before the next block. In this case, the route is activated if this has not already been done. If the activation is not completed when the train reaches the brake or stop marker in this block then the train is decelerated or stopped, respectively, in order to wait for the activation of the route. If there is only one indicator without any markers in this block, then the same indicator is used for indication of entry into the block, activation of the route and also implicitly as stop marker. In this case, the train is always stopped for a short moment because the activation of a route takes some time.

To avoid such stops it is important to use different locations in the block for brake and stop markers.

Speed Markers

A speed marker determines a point in a block, where the speed setting of the subsequent block is applied. Speed markers are represented in **TrainController™ 7 Gold** by green arrowheads. If restricted speed applies in a certain block, then the train is decelerated at the first speed marker of the previous block. If no speed marker is assigned to this previous block, then the train is decelerated at the brake or stop marker, which takes effect first.

TrainController™ assumes that a train ready to be started is located with its head near the exit of its *current block*. It is also assumed that the train will exit its current block and enter the next block just after being started. For this reason any speed conditions of the first block are ignored and the train is accelerated to the speed, which applies in the second block.

All speed changes take place at the appropriate markers of the previous block.

Action Markers

All markers described so far are also able to perform additional operations, e.g. to toggle the headlights of the passing train or to open a crossing gate, etc. These markers, however, may also change the speed of the train. If it is desired to perform operations while ensuring, that the speed of the passing train remains unchanged, action markers can be used. An action marker determines a point in a block, where operations can be performed without affecting the speed of a train. In a certain sense action markers comprise the common sub set of all other markers, namely the ability to perform operations as well as certain other properties, but unlike all other markers they do not have the built-in ability, to affect the speed of passing trains.

Action markers are represented in **TrainController™ 7 Gold** by gray arrowheads.

5.8 Arranging Indicators and Markers in a Block

B This section describes the different types of sensors and how to use them to operate a block.

Arranging Momentary Track Contacts and Occupancy Sensors in a Block

B In the following it is assumed that the track section between the turnouts in the following diagrams is a block. Several methods to arrange indicators and markers in a block are discussed below. The pros and cons of each method are outlined as well.

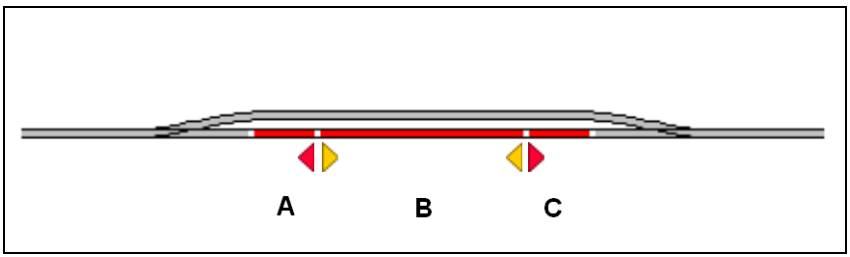


Diagram 16: Block with three occupancy sensors

Diagram 16 shows a block equipped with three occupancy sensors. Each of these sensors is associated with a contact indicator in the software called A, B and C. All indicators are assigned to the same block in the software. The block is indicated as occupied as soon as a train enters section A from the left or section C from the right. The block remains occupied until the train leaves the opposite section. A stop marker has been defined for indicator A for trains running to the left, C is marked with a stop marker for

trains running to the right. The trains are stopped at the boundary between B and A or C, respectively. The indicator B is associated with two brake markers for both directions. Trains begin to slow down when entering B from either direction. The sections A and C should be long enough, that each train is safely stopped before touching one of the turnouts. On the other side the longest train should completely fit into the block when being stopped. For this reason the boundaries between B and A or C, respectively, where trains are stopped, must be located close enough to the boundaries of the complete block.

The configuration displayed in Diagram 16 is the optimal and most recommended solution. The block is indicated as occupied as long as a train is located in one of the three occupancy sections. Additionally it would be even possible to distinguish in which of the three sections A, B or C a train is located. Lost or parking cars can be detected, too, if they cause an occupancy indication. Pushed trains can also be treated, too, if the first pushed car generates an occupancy indication. This method requires the effort, however, to cut the rails at the boundaries of each occupancy section.

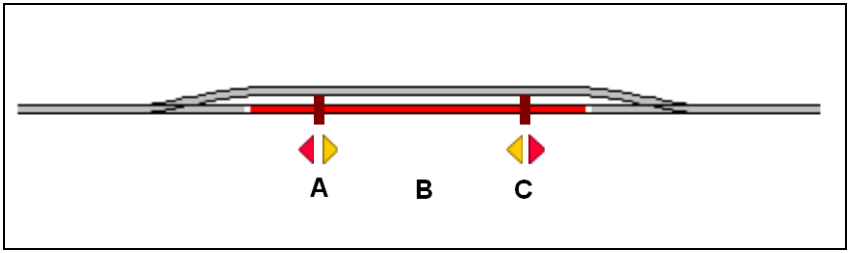


Diagram 17: Block with an occupancy and two momentary sensors

Diagram 17 shows a block equipped with one occupancy (B) and two momentary sensors (A and C). Each of these sensors is associated with a contact indicator in the software called A, B and C. All indicators are assigned to the same block in the software. The block is indicated as occupied as soon as a train enters section B from any direction. The block remains occupied until the train leaves section B. The indicator A additionally corresponds to a stop marker for trains running to the left, C is marked with a stop marker for trains running to the right. Both indicators are additionally marked with brake markers for the opposite direction, respectively. The location of A and C should ensure, that each train is safely stopped before touching one of the turnouts. On the other side the longest train should completely fit into the block when being stopped. For this reason A or C, respectively, where trains are stopped, must be located close enough to the boundaries of the complete block.

Application of Diagram 17 has take into account, that momentary contacts tend to be less reliable than occupancy sensors.

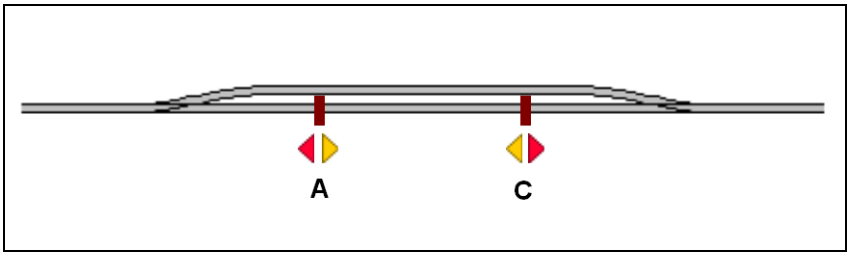


Diagram 18: Simple Block with two momentary sensors

Diagram 18 shows a simple configuration of a block equipped with two momentary sensors. Both sensors are associated with a contact indicator in the software called A and C. Both indicators are assigned to the same block in the software. The indicator A is additionally marked with a stop marker for trains running to the left, C is associated with a stop marker for trains running to the right. Both indicators are additionally marked with brake markers for the opposite direction, respectively. The location of A and C should ensure, that each train is safely stopped before touching one of the turnouts. On the other side the longest train should completely fit into the block when being stopped. For this reason A or C, respectively, where trains are stopped, must be located close enough to the boundaries of the complete block.

The configuration displayed in Diagram 18 is very simple and inexpensive but has also some disadvantages. Block occupancy is not indicated. As long as the block is reserved for a train located inside this block this causes no major problem, because the dispatcher will not allow another train to enter this block. But certain measures are to be taken to avoid premature reservation of this block for another train when a train leaves the block. There is also a disadvantage for passing trains. Let us assume that a train is passing the block from the left to the right and that a route is to be activated before the block ahead, to the right of this block. As soon as the passing train enters the block at A the route is activated. In the same moment the train begins to slow down, because A defines also a brake marker and the train has to wait, until the route is reported to be activated which needs a certain time. This could be avoided by adding an additional contact according to the following diagram:

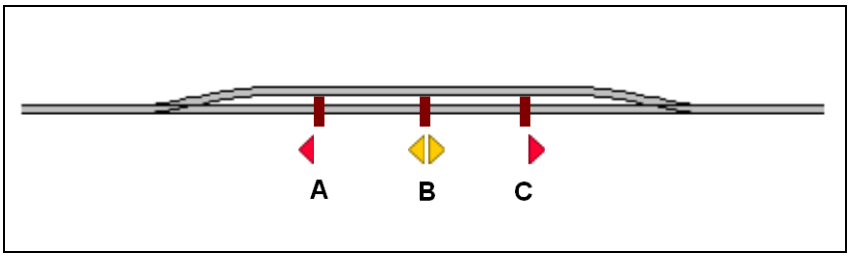


Diagram 19: Block with three momentary sensors

In Diagram 19 the indicator A defines a stop marker for trains running to the left, C acts as stop marker for trains running to the right. Indicator B is marked with brake markers for trains running in both directions. In this configuration block occupancy is not indicated, too, and as in Diagram 18 certain measures are to be taken to avoid premature reservation of this block for another train when a train leaves this block. But trains can pass this block without any speed changes, even if there is a route to be activated before the block ahead – provided the distance between A and B or C and B, respectively, is large enough, that the route can be activated after passing A or C, respectively, and before reaching B.

All examples discussed so far can be applied for blocks passed by trains in both directions. The configuration can be made simpler if trains pass a block only in one direction. This is shown in the following:

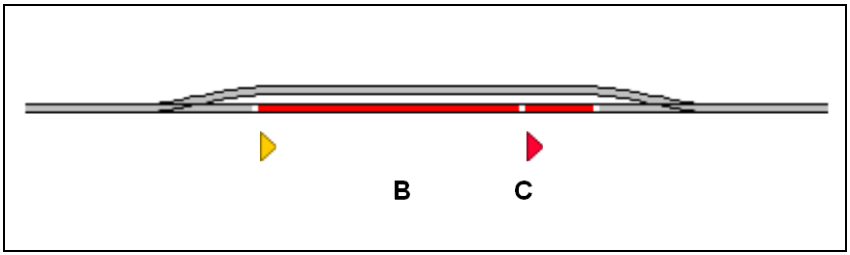


Diagram 20: Block with two occupancy sensors

Diagram 20 has been derived from Diagram 16 by eliminating sensor A. It is assumed that the block is only passed from the left to the right. B acts as brake marker and C as stop marker for trains running to the right.

The different configurations discussed in this section are only examples. Configurations similar to Diagram 20 can also be made with momentary contacts instead of occupancy sensors or with a mixture of both types similar to Diagram 17. One can think also of other configurations. There is no best way to setup a block. The optimal solution does

not only depend on technical requirements but also which equipment you already have and how much money you want to spend for new equipment.

One Sensor per Block: Shifted Brake or Stop Markers

In the examples discussed so far all locations, where trains stop or begin to brake are identical to the entry of an occupancy section or to the point, where a momentary track contact is triggered. In Diagram 19 we even installed an extra sensor to isolate the location, where the entrance into the block is reported, from the location, where the train begins to brake to gain time for activation of subsequent routes.

But it is not essential to install extra sensors for this purpose. It is also very easy to specify, that a stop marker is located in a certain distance from the point, where the associated sensor is turned on. This is done by specifying a *distance* for such marker. This creates a *shifted stop marker.*

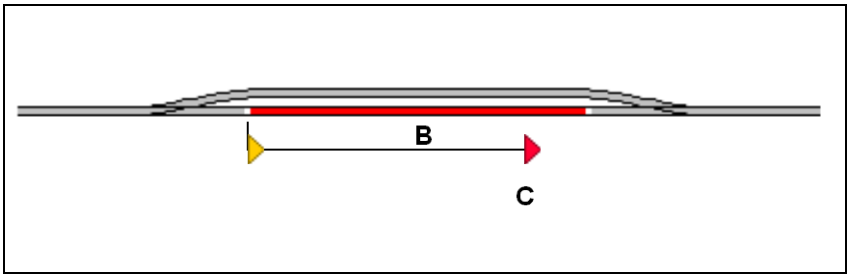


Diagram 21: Shifted Stop Marker

If your trains are running very precisely and have been profiled accordingly, then it is not necessary to install a separate sensor to mark the stop point C. Instead it is possible to mark the occupancy sensor B with a brake marker and a shifted stop marker for stop point C.

Assume that in the above example the desired stop point C is located in a distance of 50 inches from the left border of the occupancy section B. If it is desired, that trains decelerate and stop within 50 inches after entrance into B, then contact B is marked with a (shifted) stop marker with a distance of 50 inches. Additionally a brake marker should be added to B with a braking ramp of just under 50 inches to accomplish smooth deceleration.

If a train, that has to stop in this block, enters the occupancy section B from the left, it will be decelerated to threshold speed within 50 inches from the left border of B. When it arrives at the point C, which is 50 inches away from the entrance to B, the train will be automatically stopped.

In other words: the shifted stop marker associated with B works exactly like an additional sensor located 50 inches behind the entrance into section B, which is marked with a stop marker.

This principle can be certainly applied for the opposite direction, too. In this way one single occupancy sensor (sensor B in this example) can be marked with brake markers and shifted stop markers for both directions. For reasons of simplicity the markers for the opposite direction have been omitted in Diagram 21.

In Diagram 21 each train, that has to stop in this block, will begin to slow down just in the moment, when it enters the track section B. As mentioned earlier, this can cause trains to slow down temporarily, if a subsequent route is to be activated. To avoid this, it is also possible to specify a distance for the brake marker, too, which leads to a *shifted* brake marker. The principle is shown in the diagram below:

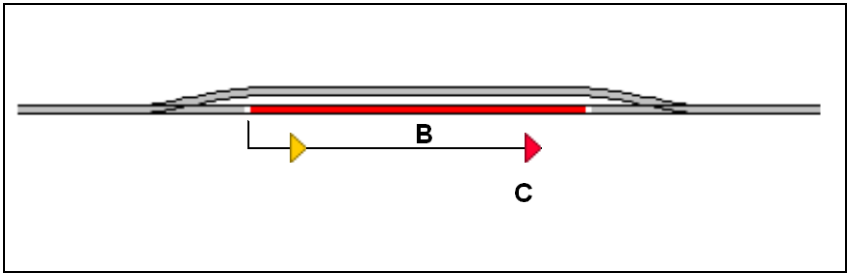


Diagram 22: Shifted Brake and Stop Markers

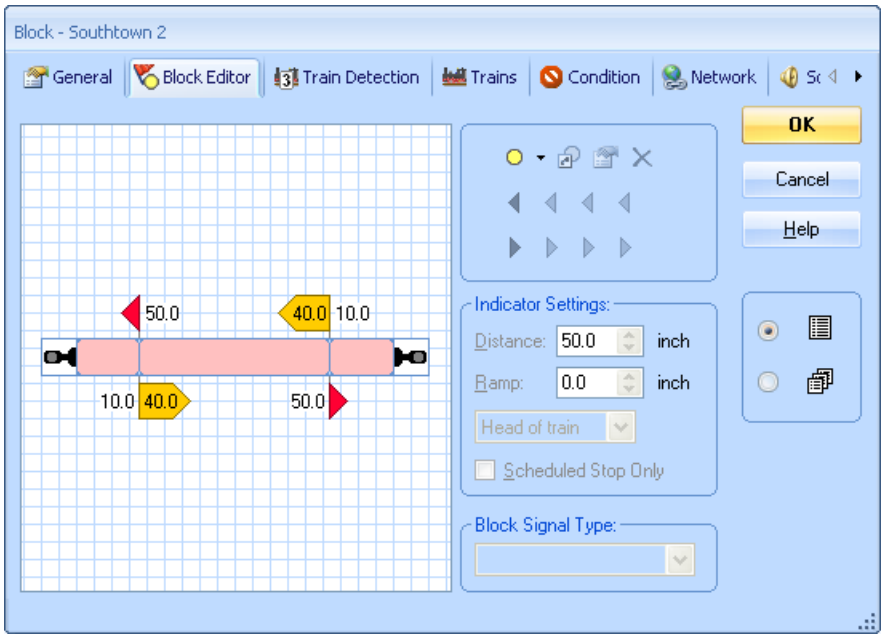


Diagram 23: Editing Shifted Brake and Stop Markers in the Block Editor

Diagram 23 demonstrates, how shifted brake and stop markers are arranged for both directions in the block editor. Trains, that shall stop in this block, will begin to slow down 10 inches behind the entrance of the occupancy section. The braking ramp is set to 40 inches, thus trains will reach threshold speed 50 inches behind the entrance, where they stop, because this is exactly the distance of the shifted stop markers.

The complete configuration displayed above can be created with the block editor for both directions in no time at all with a few mouse clicks.

! **Shifted brake or stop markers allow operation of a complete block with one single sensor and indicator symbol. Proper Functioning of shifted brake and stop markers require appropriate profiling of the affected locomotives.**

Stopping a Train in the Middle of a Platform

With the block editor it is quite simple and straightforward to accomplish a train stop in the middle of a platform.

- Create a block, that represents the track passing the platform.

- Open the block editor and create an indicator, that represents an occupancy section in this block. (If you are working with momentary track contacts and the trains shall pass the track section in both directions, then two sensors and indicator symbols are required.)
- Mark the indicator (or indicators) with a stop and a brake marker for each affected direction of travel.
- Specify appropriate distances for these markers and an appropriate ramp for the brake marker.
- Apply the option **Middle of train** to the stop marker.

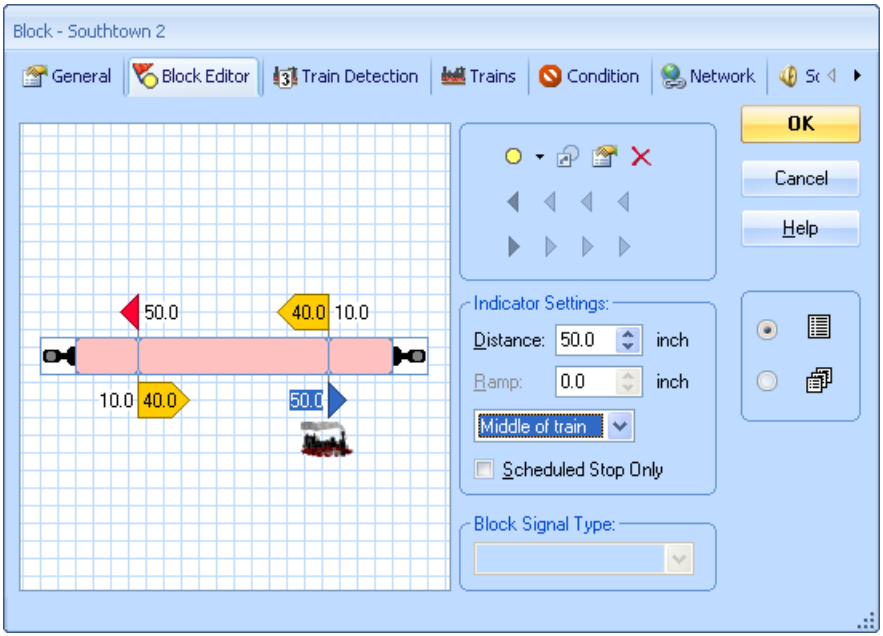


Diagram 24: Specify a shifted Stop Marker for Centred Stop

That is all. Each train, that has to stop in this block will automatically stop centred with regard to the location determined by the stop marker.

Stopping different Trains at different Positions

Passenger trains shall stop in the middle of the platform while freight trains shall advance and stop at the end of the block? This can be easily accomplished with **TrainController™ 7 Gold**, too:

- Add a further stop marker to the configuration described in the previous section.
- Apply **Head of train** to this additional marker.
- Select this stop marker and call the **Properties** command in the toolbar located in the upper right corner of the block editor.
- Select the (freight) trains, to which the additional stop marker shall apply.
- Repeat the last two steps for the stop marker created in the previous section, which is responsible for stopping trains in the middle of the platform, and specify the (passenger) trains, to which this marker shall apply.

If this has been done correctly all passenger trains will automatically stop centered with regard to the location determined by the first stop marker, while freight trains will advance and stop at the position defined by the second block marker.

It is not only possible to specify, that a certain marker is only valid for specific trains, it is also possible to specify, that a certain marker is only triggered in conjunction with certain schedules. These schedules are specified in a similar way like the associated trains. Limiting markers to specific schedules is useful, for example, if the same train shall stop at different locations depending on the currently executed schedule. It can be also useful in cases, where it is simpler to select a few schedules rather than a plurality of trains in order to specify different stop points for different trains.

If a certain marker is limited to specific trains and to specific schedules, then the marker is only triggered, if one of these trains passes this marker under control of one of these schedules. The marker remains turned off, if the train does not match or if the schedule does not match or both.

It is possible to define an arbitrary number of stop, brake and speed markers in a block. It is also possible to specify, that a certain marker takes effect only under certain conditions. In this way it is for example possible to define different stop points for different operational situations.

The features described above and their combinations provide virtually unlimited possibilities to determine, where trains shall stop, slow down or change their speed in the particular blocks.

Markers for scheduled Stops vs. Markers for unscheduled Stops

It is possible to specify differing brake and stop markers for scheduled stops and unscheduled stops in the same block.

Markers specified for scheduled stops are only triggered, if the train has to perform a scheduled stop in this block. If the train stops for another reason, e.g. because the exit of the block is currently locked, then such marker is not triggered.

If at least one stop marker in a block is specified for scheduled stops, then all other stop markers in this block, which are not specified for scheduled stops, are only triggered for unscheduled stops. This applies accordingly to brake markers, too.

This feature can be used to specify different stop points for the same train in the same block, which depend on, whether the train has to perform a scheduled stop or an unscheduled stop in this block. A good example is a passenger train, which shall sometimes perform a scheduled stopover in the middle of a certain block, and which shall pass this block without scheduled stop otherwise. This can be accomplished by specifying a stop marker located in the middle of the block for scheduled stops and another stop marker at the location of the block signal for unscheduled stops. A passing train, that has to perform a scheduled stop in this block, will stop in the middle of this block. In other cases, when this train does not have to perform a scheduled stop, but must stop here for any other operational reason, the train will stop at the block signal.

5.10 Run Trains with Interlocking

After arranging the block system in **TrainController™ 7** as outlined in the previous sections it is possible to run trains under full protection and routing of the computer. Put a locomotive on the track, assign its symbol to the associated block and call the menu command **Run with Interlocking**. The train will immediately start to move, provided that the route ahead is clear. It will then select an appropriate path and continue to travel, until it reaches a dead end or until the path ahead is blocked for another reason. At a dead end it will reverse automatically, if desired, and continue to travel to the opposite direction.

With this method routes can be treated in different manners. It is either possible to allow the computer to select and activate all routes requested by the train automatically. It is also possible to leave this to the human operator. In this case the train is stopped in blocks with at least one outgoing route, until one of these outgoing routes is selected and activated by the human operator.

If no further measures are taken, trains may run anywhere on your layout. By permitting certain blocks for certain trains only it is possible to direct trains to specific locations. Another possibility to gain control over the path each train is taking is the prior definition of schedules. This is outlined in the following.

5.12 Execution of Schedules

Preset Block Signals and Speed Limits

As outlined in section 5.9 **TrainController™** automatically calculates signal aspects for all trains running under control of the *Dispatcher*. These signal aspects take into account the availability of blocks and routes ahead of the train. If the train must not enter a certain block, then the concerning signal of the previous block is set to "red". If the train may enter the block, then the signal is usually set to "green". It is additionally possible, however, to cause **TrainController™** to display "yellow" in case "of green", if desired.

For this purpose it is possible to select an individual signal aspect (yellow or green) for each block or each route in a schedule. Depending on this setting **TrainController™** will automatically apply the selected color to the calculated block signal, if the train may proceed.

These signal settings are specified on the level of blocks and routes in a schedule. That means: the same block or route may have differing signal settings in different schedules.

It is additionally possible to adapt the train speed to the selected signal aspect. This is done by specifying speed limits for the green and yellow signal aspect in the properties of each block. For each block it is preset in this way, at which maximum speed each train may pass this block depending on the currently selected signal.

Assume a block with the maximum speed (green signal) set to 80 mph and the restricted speed (yellow signal) set to 30 mph. If the signal for this block in schedule "A" is set to green, then the train will pass this block with 80 mph, when schedule "A" is executed. If the signal for this block in schedule "B" is set to yellow, then the train will pass this block with 30 mph, when schedule "B" is executed.

The available speed limits for the green and yellow signal aspect are specified globally on the level of each block. In a schedule it is then selected, which of the two speed limits applies for this block in this schedule.

> The above describes the default policy. **TrainController™ 7 Gold** provides even more possibilities to adapt the calculated block signals and applied speed limits to personal needs.

TrainController™ 7 Gold it is not only possible to select the desired signal indication (green or yellow) for each block or route in a schedule, it is also possible to specify once for all schedules, that the calculated block signal of certain blocks or routes shall be yellow. And even more: it is possible to preselect the yellow signal indication individually for each particular position of a turnout. The signal indication specified for a specific turnout position is then accordingly propagated to all routes, that contain this turnout in this position.

The hierarchy of the various signal settings is as follows: to calculate the block signal indication for a certain route or block in a schedule, **TrainController™ 7 Gold** at first checks, whether the yellow signal has been selected for this block or route in the settings of the schedule. If this is not the case, then the properties of the block or route are checked instead. In case of a route the preselected block signals for the according positions of all turnouts contained in this route are checked, too. If at least one of the checked objects requests a yellow block signal, then the resulting block signal is yellow, too. In all other cases the resulting block signal is green.

The possibility to preselect signal indications on the level of blocks, routes or turnouts provides several advantages:

- In cases, where a specific block or route shall always be passed with the same signal indication, it is much more convenient to preselect the indication once for all schedules in the properties of the block or route.
- The possibility to preselect a signal on the level of turnouts is useful, if specific turnouts shall always be passed with the same signal indication. It is much more convenient to preselect this indication once for all routes and schedules, which use this turnout, in the properties of the turnout, rather than being forced, to repeat the same selection in all affected routes or schedules.
- The possibility to specify differing signal indications for the particular positions of a turnout is useful, if the indication of the calculated signal shall depend on the turnout position.
- The signal indication preselected for a block, route or turnout is also applied to trains run by **AutoTrain™**. In other versions of **TrainController™** the signal indication for trains run by **AutoTrain™** is always green in all blocks or routes and cannot be changed.

As outlined earlier there is a close connection between calculated block signals and speed limits. The allowed speed of a train depends on the currently valid signal indication (green or yellow). While other versions of **TrainController™** only allow to specify speed limits on the level of blocks, **TrainController™ 7 Gold** allows to specify speed limits also on the level of schedules, routes and even turnout positions. The hierarchy of the various settings is similar to that of the signal settings described above. To

calculate the speed limit for a certain route or block in a schedule, **TrainController™ 7 Gold** at first determines the speed limit specified for this block or route in the settings of the schedule. Additionally the speed limit preset in the properties of the block or route is determined, too. In case of a route the speed limits for the according positions of all turnouts contained in this route are determined, too. The final speed limit results from the minimum of all determined limits. If no speed limit is specified for a certain object in this chain, then the settings of this object do not affect the resulting speed limit.

The possibility to preset speed levels on the level of routes or turnouts provides the same advantages as listed above. Especially it is possible with **TrainController™ 7 Gold** to propagate speed limits valid for a certain position of a turnout to all affected schedules and also to trains run by **AutoTrain™**.

Temporary Speed Limits

A temporary speed limit can be accomplished by executing a specific train operation, e.g. by a marker in a block. Temporary speed limits can only be applied for trains running under control of a schedule, AutoTrain or interlocking.

If the current speed of the train, to which the speed limit is applied, exceeds the specified value, then the speed of the train is reduced to the specified speed as soon as this operation is executed. If 0 is specified as speed, then an effective speed limit, if any, is cleared. When a train terminates a schedule, then an effective speed limit, if any, is automatically cleared, too. This is also true, if control of the train is passed to a successor schedule. Temporary speed limits are only effective in the scope of the current schedule of the affected train.

Wait Time

Additionally you can specify a *wait time* for each block contained in a schedule in order to perform scheduled stops in certain blocks of a schedule.

In **TrainController™ 7 Gold** it is furthermore possible to specify an individual delay for each scheduled stop. Such delay is applied after a scheduled stop has ended, while associated operations are executed and before the train is set in motion. This time span can be utilized to perform additional operations (e.g. playing an announcement, the noise of closing doors or the whistle of the conductor) after a scheduled stop ended and before the train is set in motion (see below).

Additional Operations

Finally it is possible, to assign Operations to each block of a schedule. Possible operations are turning on or off an *engine function* (see section 3.6, “Headlights, Steam and Whistle”), execution of certain train operations or execution of a list of operations in order to perform more complex actions.

These operations can optionally be performed when

- the train enters the block
- the train reaching a brake indicator has to reduce its speed
- the train has to stop
- the train starts again after a stop
- the block is released after the train has left the section

Additionally it is possible to perform operations before starting or after finishing the schedule.

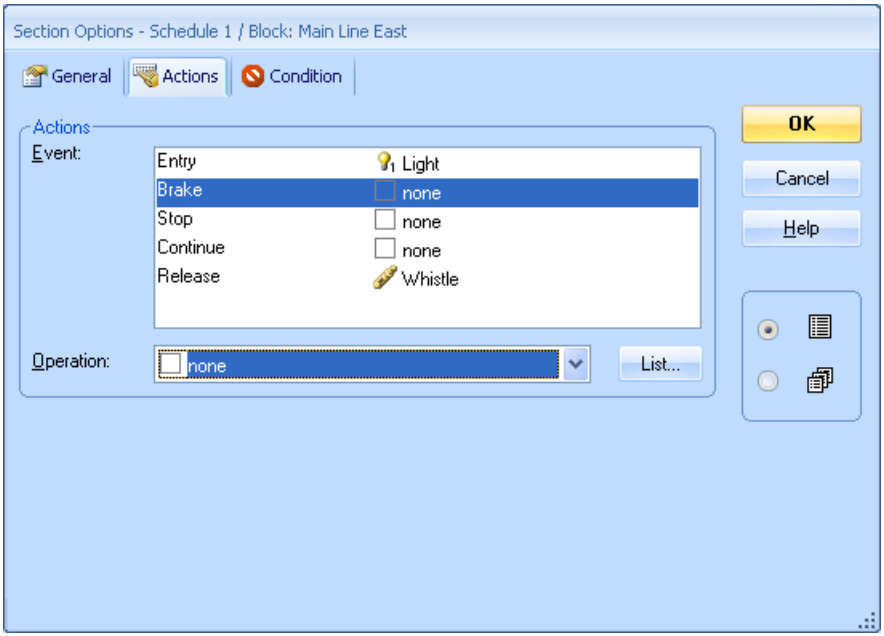


Diagram 25: Specifications of the Section of a Schedule

In the example displayed above each train entering the related block shall turn on the light. Additionally it shall blow its whistle when the block is released later.

If a function symbol specified here is not configured for an engine, then this engine will do nothing, when it executes this schedule. If, for example, the function symbol *Whistle* is only assigned to steam engines in the example displayed above, then diesel engines will remain quiet when executing this schedule.

These attributes are specified on a per-schedule base. It is possible to specify different settings for different schedules.

Running Trains manually under Control of a Schedule

For each schedule you can specify its *driving mode*. If desired you can control engine and trains on the schedule completely manually. In this case the computer reserves the blocks, activates the routes and calculates the block signals. You are – like a real engineer – responsible for obeying the indicated signals and following the speed conditions. But it is also possible to transfer the control over the schedule completely to the computer. In this case all engines and trains on this schedule are operated automatically. Finally it is also possible to share the engineer's job with the computer. In this way it is for example possible, that the train is running under your manual control, but that the computer is able to intervene to stop a train in front of a red signal.

Driving Mode	Explanation
	Trains are completely controlled by the computer
	The computer intervenes when restricted speed is prescribed or when the train approaches a red signal requesting the train to stop.
	The computer intervenes, when the train approaches a signal requesting the train to stop.
	Trains are almost completely controlled manually. If the human operator fails to stop the train in time before reaching the stop marker in front of a red signal, then the computer performs an emergency stop of the train.
	Trains are completely controlled manually.

Table 1: Driving Modes of a Schedule

It is possible to use different modes for different schedules, regardless whether these schedules share the same blocks and routes or not. This enables full automatic operation

of one part of your layout and running trains manually under computer control in another part.

Different schedules with different modes can be arranged for the same part of your layout, too. It is for example possible to create two schedules for the main track of your layout. The first schedule is used for automatically running trains, while the second schedule uses the same track for trains operated manually under control of the computer. In this way you can operate your favorite train manually while other trains in front of or behind this train are controlled automatically.

Driving modes can also be specified individually for each particular engine. If this is done, then the driving mode of the engine overrides the setting of the schedule. This is useful if you want to run different trains in different driving modes with the same schedules.

5.13 AutoTrain – Start of Schedules made Easy

B

AutoTrain™ is another outstanding feature of **TrainController**™. With *AutoTrain*™ you can run automatic trains at any time during operation without the need to define schedules before.

AutoTrain™ is especially useful in the following cases:

- If a train shall automatically run somewhere during operation and you did not specify an appropriate schedule before to perform this task.
- If you want to define a new schedule quickly from scratch.

Auto Train by Drag & Drop

The fastest way to run *AutoTrain*™ is Drag & Drop with the mouse:

- Open the **Schedule** menu and call the **AutoTrain by Drag and Drop** command (alternatively press and hold the key 'A' on your computer keyboard; A = *AutoTrain*™).
- Press the left mouse button near the exit of the block in the block diagram or in the switchboard, where the train shall start.
- Hold the left mouse button pressed and drag the mouse to the exit of the block in the block diagram or in the switchboard, where the train shall stop.
- Release the left mouse button.

- Call the **AutoTrain by Drag and Drop** once more to turn this mode off (or release the 'A' key).
- The train will now start and run automatically to the destination block.

Auto Train Toolbar

With the *AutoTrain*™ toolbar you have more options for individual customization before the train is actually started. To run a train with the *AutoTrain*™ toolbar the following steps are performed:

- Open the *AutoTrain*™ toolbar .
- Select the locations (blocks) on the layout, where the train shall start.
- Select the locations (blocks) on the layout, where the train shall stop.
- Optionally specify additional options that influence the execution of the *AutoTrain*™, such as wait time, operations, cycle, shuttle, etc).
- Start *AutoTrain*™.

Diagram 26: AutoTrain Tool Bar

After start *AutoTrain*™ automatically tries to find a path from the specified start block to the specified destination blocks. If a train is located in the start block, then it is automatically started to run to the selected direction.

A started *AutoTrain*™ is very similar to a schedule which is currently executed. It has one starting block and one ore more destination blocks, that are selected before *AutoTrain*™ is started.

There are some additional options:

- After selection of the start and destination blocks you can let *AutoTrain*™ try to find a path from the start to the destination blocks without starting a train. This is useful in *edit mode*, especially if no train is located in the start block. This is also useful if you want to check the resulting path before actually starting the train. Together with another option, that allows to store the current *AutoTrain*™ as a permanent schedule for later use, this is a very fast method to create new schedules by letting the software calculate the appropriate paths for you.
- It is possible to select certain blocks or routes to be included in the schedule anyway prior to start the search for an appropriate path. Each found path will then pass these blocks or routes, if possible. This gives you more control over the resulting path.

- It is also possible to exclude certain blocks or routes from *AutoTrain*™ prior to start the search for an appropriate path. This gives you additional control over the resulting path, too.
- You can also specify, whether only the shortest possible paths from the start to the destination blocks shall be taken into account or all possible paths.
- Additionally it is possible to limit the search time. This option is useful in case of large or complex layouts and slow computers, where the search may take a certain while.

While an *AutoTrain*™ is active you can also store it as a schedule to execute it later, e.g. as part of a time table.

AutoTrain with Start and Destination Keys

In **TrainController™ 7 Gold** it is also possible to perform *AutoTrain*™ as operation of other objects. This is in particular useful, if you want to start *AutoTrain*™ with push button symbols by using them as start and destination keys.

AutoTrain™ operations are always associated with certain blocks.

AutoTrain™ operations do not distinguish between the start and the destination of the run. If two *AutoTrain*™ operations are called for Block A first and then for Block B, then the train runs from A to B. If the same operations are called in the opposite order, then the train runs from B to A. *AutoTrain*™ operations should be always called in pairs. The first operation determines the start block and the direction, in which the train starts. The second operation specifies the destination block and the direction, in which the train enters the destination block. The second operation also starts the train.

You can use *AutoTrain*™ operations in a macro, for example, to run a train from one block of your layout to another. In this case you should ensure, however, that two *AutoTrain*™ operations, one to specify the start block and one to specify the destination block, are contained in the macro.

An interesting use of these operations and the actual reason, why these operations are provided, is the use with push button symbols as start and destination keys. To accomplish this, assign one *AutoTrain*™ operation to the operations of each related push button. *AutoTrain*™ operations should be executed in pairs. This is accomplished during operation by pressing one push button with such operation assigned and then another. The first push button determines the start block and the direction, in which the train starts. The second push button determines the destination block and the direction, in which the train enters the destination block. The second push button also starts the train.

Thanks to the fact, that *AutoTrain™* operations do not distinguish between start and destination, the same pair of push buttons can be used, too, to start a train in the opposite direction. It is the order, in which the operations are executed, which is relevant for the determination of start and destination, not the operations themselves.

AutoTrain™ with start and destination keys is useful to run trains in a default manner, i.e. without any specific additional actions, on point to point connections. No prior definition of schedules is necessary, which keeps the schedule list less populated. By assigning *AutoTrain™* operations to the operations of feedback indicator symbols, which are again associated with push buttons on external control panels, it is even possible to trigger *AutoTrain™* from such remote panels.

5.14 Schedule Sequences

With schedule sequences a series of schedules can be executed in **TrainController™ 7 Gold** one after the other with the same train. Schedule sequences contain a list of other schedules. When the sequence is started, then the first schedule in the list is started, too. After termination of the first schedule the second schedule in the list is started with the same train and without stopping the train, if possible. After termination of the second schedule in the sequence the third schedule is started and so on, until the last schedule in the list is completed.

With schedule sequences it is possible to create long schedules by combining several shorter schedules. Sequences are for example useful to create a plurality of long schedules with a library of several short schedules as building blocks.

5.15 Successors of a Schedule

For each schedule it is possible to specify a set of successor schedules, of which one is started, after the schedule is finished.

Several options allow you to specify how control of the train is passed from a schedule to its successor:

- The successor can be selected **by order** or **randomly**.
- Additionally you can select to **keep the train**, i.e. to enforce that the successor continues with the same train as before, or to enforce a **train change**.
- **TrainController™ 7 Gold** also allows to specify a certain car, an engine or a train group. If a car or an engine is specified, then the successor is started with the engine

or a train, that contains this vehicle. If a train group is specified, then the successor is started with a train, to which the train group applies (see also page 95).

- It is additionally possible to specify, that the successor schedule shall be started with the **oldest train**. The oldest train is the train, which was not operated by a schedule for the longest time. **TrainController™ 7** allows to combine this option with the other options. If this option is combined with the option to perform a train change, then the successor is started with the *oldest* train, that differs from the previous train. If this option is combined with the specification of a train group, then the *oldest* train, to which the train group applies, is started.
- It is also possible to specify that **all** listed successors are started. These successors are started simultaneously, when the previous schedule is about to be terminated.

Schedule Sequences vs. Schedule Successors vs. Long Schedules

Shall complex train runs be specified as schedule sequences, as a chain of schedule successors or as one complex schedule? The answer to this question depends on the individual case and is also a question of personal taste.

A train run, for example, that starts in the hidden yard of the block diagram displayed in Diagram 11, passes both blocks in “Southtown” and ends again in the hidden yard can be specified as a schedule sequence with three or four schedules, as a chain of schedule successors or as one big schedule which alternate paths in “Southtown”.

These are the pros and cons of the particular approaches:

Schedule Sequences:

- Only available in **TrainController™ 7 Gold**.
- No change of trains is possible between two subsequent schedules in the sequence.
- Usually used as a replacement for a single more complex schedule. Schedule sequences can be used to create a plurality of longer, more complex schedules by using several shorter, less complex schedules as building blocks.
- No static linkage between a schedule and the subsequent schedule in the sequence. A schedule can precede different subsequent schedules in different schedule sequences.
- The look ahead to select an optimal path out of several possible alternate paths is limited by the end of the current schedule in the sequence. This can improve the performance of the path selection, but may lead to selection of non-optimal paths.
- Schedule sequences can be started in reverse direction, i.e. beginning with a destination block of the last schedule in the sequence end ending in a start block of the first schedule in the sequence.

Schedule Successors:

- Change of train is possible between two subsequent schedules in the chain of successor schedules.
- Static linkage between a schedule and the subsequent schedules in the chain of successors. A schedule always acts as predecessor of the same successors.
- Like schedule sequences the look ahead to select an optimal path out of several possible paths is limited by the end of the current schedule with the same consequences as with schedule sequences.
- Chains of schedule successors cannot be started in reverse direction.

Single long Schedules:

- Change of train is not possible, until the schedule is terminated.
- The look ahead to select an optimal path out of several possible alternate paths can take into account the complete path to the block, where the train shall finally stop. This supports the selection of optimal paths at the expense of program performance.
- Single schedules can be started in reverse direction, i.e. from a destination block to a start block.
- Single schedules can be repeated as cycle or as shuttle train.

Conclusions:

- If you want to change the running train between two subsequent schedules, then both schedules must be chained as successors. This is for example useful, if a train entering a hidden yard shall trigger another train to leave this yard.
- If the train shall not be changed, then it is usually better to create a schedule sequence (**TrainController™ 7 Gold** only) or a complex, long schedule rather than a chain of schedule successors.
- If a schedule shall be repeated as cycle or commuter train, then use a single schedule for this purpose.
- Critical sections cannot span different schedules. They must be completely contained in the same schedule.
- If a single schedule becomes very complex or long, then consider to split it into a schedule sequence of several more simple schedules.
- If the track plan of your layout allows to derive a plurality of more complex schedules from a relatively small set of simple basic schedules, then consider to create the complex schedules as sequences of these simple schedules.

It is possible, though not very recommended, to mix schedule successors and schedule sequences. This is treated by the software in the following way: chaining by successors has higher priority than chaining by sequences. That means: if schedule B is specified as successor of schedule A, and schedule A and a third schedule C are listed consecutively in a schedule sequence, then A is executed first by this sequence, then B (as successor of A) and finally C (as the second member of the sequence).

5.17 Operation Interruption - Termination of Schedules

There are several methods to interrupt the running operation or to terminate schedules. These methods can be accessed by different menu commands. They are described in the following:

- Global **Stop**: This command performs an emergency stop of all connected digital systems and terminates all running schedules. This is the most drastic method to terminate operation and should only be used in very rare, extreme emergency cases. Since all schedules are terminated the computer releases control of all previously running trains. If the emergency stop of the connected digital system is released later, then the software does not have control over any trains.
- **Freeze**: This command performs an emergency stop of all connected digital systems and interrupts all running schedules. This is the recommended method to stop operation in emergency cases. The software keeps control over all previously running trains. After resolving the emergency situation and clearing the freeze state of the software the operation can be continued at that position, where it has been interrupted before.
- **Stop Train**: This command stops the selected train abruptly, but does not terminate any running schedule. It can be used to clear an emergency case, where only one single train is affected.
- **Stop All Trains**: This command stops all trains abruptly, but does not terminate any running schedule.
- **Terminate Schedule / Run**: This command stops the selected train abruptly and terminates its current schedule or run with interlocking, respectively. It can be used to terminate a running schedule prematurely or to stop a train running with interlocking.
- **Terminate all Schedules**: This command stops all trains abruptly and terminates their current schedules or runs with interlocking, respectively.
- **Lock all Blocks:** The methods listed above stop the affected train always abruptly. To stop trains smoothly additional measures must be taken or custom methods must be configured by the end user in versions other than **TrainController™ 7 Gold**. In **TrainController™ 7 Gold**, however, this method can be used to interrupt the oper-

ation of the layout by stopping all trains smoothly at the next appropriate location. If this method is applied, then all currently not reserved blocks are prevented from being reserved. Each train, which is currently controlled by a schedule or which is running with interlocking, will process the blocks, which it has already reserved and will then stop smoothly without reserving any additional blocks.

- **Lock all Schedules:** This method can be used to terminate the operation of the layout while stopping all trains smoothly at the destination of their current schedule. If this method is applied, then all schedules are prevented from being started. Each train, which is currently controlled by a schedule, will process to the next destination block of its current schedule and will then stop smoothly without starting any additional schedules.

Especially the last two methods, which are available in **TrainController™ 7 Gold** only, are well suited to interrupt or terminate operation without causing trains to perform an abrupt stop. If **Freeze** is applied additionally after all trains have smoothly come to rest, it is possible to terminate the operating session completely and to start the next session again at another day and at the same position.

5.18 Putting all together – The Dispatcher Window

The *dispatcher window* serves as display for the block system of your layout. It lists and displays all diagrams, blocks, routes and schedules.

TrainController™ 7 Gold allows to open as many dispatcher windows as you like. It is for example possible to open a separate dispatcher window for each existing block diagram. By grouping a dispatcher window together with a switchboard window within the same parent frame window (either docked side by side or tabbed) it is possible to create several grouped windows, that contain a switchboard each together with the associated block diagram. Especially if these windows are tabbed it is possible to toggle conveniently between the switchboard view and the block diagram view of the related part of your layout.

The dispatcher window is split into two parts. The left part lists the blocks, routes or schedules of your layout. With the particular controls of the page navigator it is quite easy to switch from one view to another. Depending on the selected view additional detail information is also available. The block and route view, for example, additionally provide an optional view of the indicators and markers contained in the currently selected block or route. The schedule view provides optional lists of the blocks and routes contained in a selected schedule and allows also to display a view of all indicators and markers, that are contained in a certain block or route of this schedule.

The optional indicator view furthermore provide another interesting feature: in offline mode, i.e. if the indicator symbols in the view are currently not connected to an actual digital system, it is possible to turn the status of these symbols on and off by clicking to them with the left mouse button. In this way the sensor events generated by passing trains can be conveniently simulated.

The right part of the dispatcher window displays the currently selected block or schedule diagram. It is possible to switch from one diagram to another by using the diagram selector menu in the upper right corner of the dispatcher window. Click to the name of the current diagram, which is displayed by well visible letters in the upper right corner of the dispatcher window to open the menu of available diagrams and to change to another diagram.

All routes displayed in the dispatcher window, regardless whether displayed in the route list or in the block diagram, can be operated with mouse click, too, when edit mode is turned off.

5.19 Customizing the Dispatcher Window

General

The dispatcher window can be freely resized and zoomed. This allows to let the display fit the dimensions of the displayed block diagram optimally.

The colors of the window background, blocks and connecting routes can be adjusted to personal taste, too.

The display of block signals and train images can be turned on or off.

Additionally to these general customization features, which have been already available in previous versions of the software, **TrainController™ 7** provides the following additonal customization features:

- A new option allows reset of all display options to factory defaults.
- It is optionally possible now, to display the names of blocks in the block diagram, too, when edit mode is turned off. Previous versions of the software displayed block names solely in edit mode.

- Active routes can be displayed with individually specified colors (as in previous versions), or with the color of the reserving train, if any, or with a color, that is common for all active routes.
- The highlight color of occupied routes can be controlled by the reserving train, if any, as in previous versions, or by the color of the occupied indicator or by specifying a constant color value.
- The display intensity of blocks and routes, that do not belong to the currently selected schedule can be dimmed to fit personal taste and to support low contrast display environments.

Display of Schedules

The visibility of schedules in the dispatcher window can be limited during operation to those schedules, that you actually want to see listed then. This is controlled by the option **Visibility in Edit mode only** in the properties of each schedule. If this option is turned on, then the schedule does not occur in the schedule list in the dispatcher window, when edit mode is turned off. This is for example useful, if you want to exclude those schedules from being listed, that are started as successor of other schedules or by start/destination keys or if you want to limit the list to those schedules, that you want to start explicitly after selecting them from the list.

Another option of **TrainController™ 7**, that is related to the display of schedule diagrams, controls the intensity of the optional background display of those blocks and routes, that are not included in the schedule. This intensity can be customized to ease the differentiation between included and not included blocks and routes, which is in particular useful for low-contrast computer screens.

6 The Traffic Control

The design of the traffic control has been revised in **TrainController™ 7**, too. The layout is much more intuitive than before and some additional options are also available.

In **TrainController™ 7** **Gold** it is furthermore possible now, to open as many traffic control windows as desired. Other versions of the software are limited to display of only one traffic control window at a time.

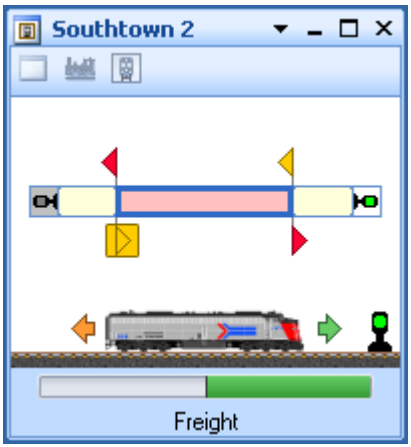


Diagram 27: The new Traffic Control

The traffic control now displays the status of the block and its indicators, the stop, brake and speed markers as well as the orientation and direction of travel of passing trains in a much more intuitive manner.

If the digital system, to which the contact indicators belong, is running in offline mode, then you can toggle the state of each indicator by clicking to it with the mouse. In this way the movements of trains can be simulated very conveniently: simply select the block that you want to look at on the computer screen and click to the occupancy, brake or stop indicator to simulate what happens if a train passes this indicator.

The following options are additionally available in **TrainController™ 7** **Gold**:

- **Pin to current Window:** With this option the traffic control becomes associated to the currently active window. Even if that window becomes inactive the traffic control will only display objects, that are selected in this window. This option can be used to permanently watch the status of objects, that are selected in a certain window. If a traffic control is pinned to a certain train window, for example, then this traffic control will display only those trains, that are driven by this train window. This is for example useful, if a traffic control is grouped together with a certain train window (e.g. docked side by side or tabbed). In this way it is possible to create a "super" train window, that contains a regular train window combined with a traffic control, that always shows the status of the train, which is currently selected in the train window.

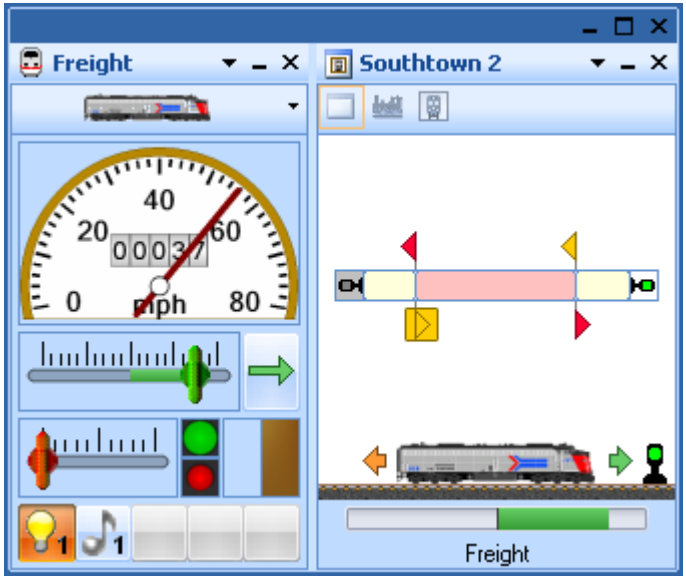


Diagram 28: Grouped Train Window and Traffic Control

- **Pin to current Train:** With this option the traffic control becomes associated to the currently selected train. Even if another object is selected this traffic control will continue to display the status of this train. This option can be used to permanently watch the status of a certain train.
- **Pin to current Block:** With this option the traffic control becomes associated to the currently selected block. Even if another object is selected this traffic control will continue to display the status of this block. This option can be used to permanently watch the status of a certain block (e.g. the entrance to a station).

9 The Simulator

With **TrainController™ 7** it is possible to simulate the operation of a model railroad automatically and without human intervention.

The traffic control (see chapter 6, “The Traffic Control”) allows to simulate the movement of running trains by triggering of the contact indicators, that belong to the particular blocks. Simulated triggering of the contact indicators is accomplished by clicking to the particular indicators with the mouse.

The simulator window can run such simulation automatically without the need for manual clicks to indicators. To start the simulation open the simulator window via the **Window** menu and press the **Start** button in the simulator window.

Diagram 29: The Simulator Window

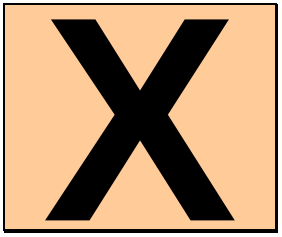
The following prerequisites must be fulfilled to run the simulation:

- The software must run in offline mode, i.e. the computer must not be connected to a digital system.
- The software must run outside the edit mode.

The simulator solely simulates the triggering of indicators by running trains. It does not operate anything. In particular it does not affect the speed or direction of running trains directly nor does it start or stop any trains. The speed of trains is set by the usual means – e.g. by running schedules or by using the controls of the train window. For running trains, however, the simulator is able to calculate, which contact indicator will be triggered next and when. These calculations are based on the current position of each running train and the path, which it is about to take.

Part III

Extensions



10 Advanced Train Control

10.1 Cars and Train Sets

Cars

Cars represent vehicles of your model railroad, that are not equipped with a motor. Examples are passenger cars or freight cars. For each car you can specify the following attributes among others:

- digital decoder address, if the car is equipped with a function decoder
- name and image
- length and weight
- maximum allowed speed
- auxiliary functions (e.g. light, smoke or coupler)

Cars are mainly used for trains, that change their formation during operation, and to accomplish the following tasks with these trains:

- the maximum speed of a certain locomotive shall vary and depend on the pulled cars (e.g. fast passenger train vs. slow freight train, both pulled by the same engine at different times)
- the same locomotive shall be directed to different tracks according to the cars it is currently pulling (e.g. passenger train may go to the platform track while a freight train pulled by the same locomotive at another time must not go there)
- trains shall be directed to different tracks according to their current length (see also page 111)
- trains shall be always able to stop in the middle of a block (e.g. a platform) even if they pull trains of varying length
- the tonnage of trains shall be simulated realistically according to the current weight of the cars contained in the train

Cars are mainly needed to accomplish the purposes outlined above and in general advanced operation of your model railroad. For reasons of simplicity novice users should postpone the usage of cars until they are familiar with the program.

Even though it is possible to create a car record in TrainController™ 7 Gold for each particular car of your model railroad it is recommend to go with as few cars as possible.
If a certain train does not change its formation during operation, then it is sufficient to create a simple engine record for such train. Engines also have a length and a weight and if these attributes do not change during operation you can specify the length or weight of the complete train pulled by this engine in the properties of the engine, too.
If a certain set of cars is always coupled together but pulled by different engines, then one should create only one single car object in TrainController™ 7 for this set and assign an appropriate name, image, length and weight to it, which represent the according attributes of the complete set of cars.

Train Sets

A train set is composed of a couple of engines or cars. Train sets can be created, arranged and dissolved at any time during operation of the layout.

Train sets are not only used to operate cars together with engines to accomplish the purposes outlined in the previous section. Train sets are also used to accomplish realistic *multiple unit* operation, i.e. operation of trains, that contain more than one engine.

Similar to real railroads each single engine, which is operated separately, or each car, which is located isolated on your layout, can also be seen as a train. For this reason the term train is usually used in **TrainController™ 7** as a generic term for each engine, isolated car or complete train set.

Train sets can be arranged via the Train List by dragging the symbols of engines and cars with the mouse while pressing the Alt-key on your computer keyboard. They can also be arranged with the Train Set dialog box displayed below:

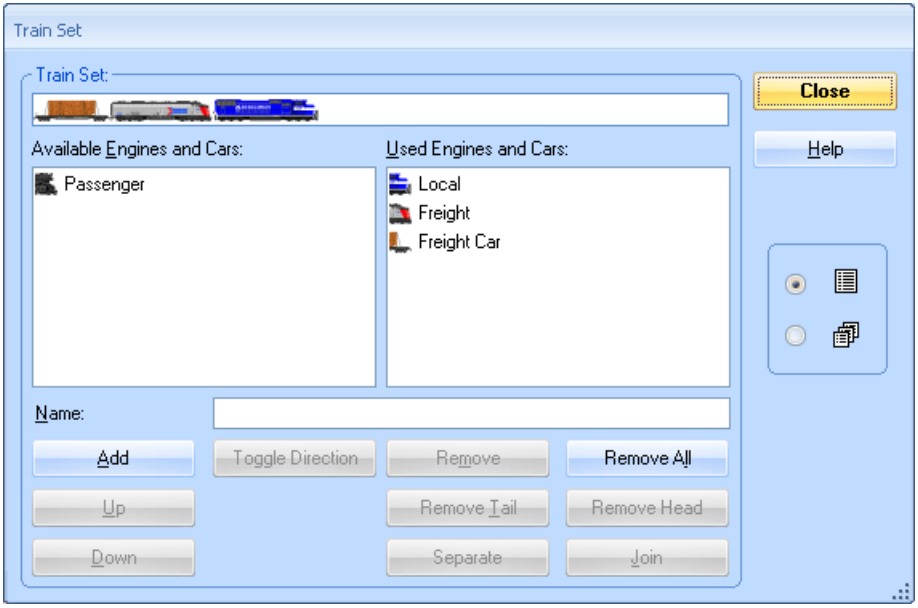


Diagram 30: Arranging a Train Set

The options of this dialog allow to add engines or cars to a train set, to remove vehicles, to change the direction of a vehicle within the train set or to split train sets into two other train sets.

To operate the speed or direction of a train set select an arbitrary engine currently contained in this train set in the Train Window. Changing the speed or direction of this vehicle will also be applied accordingly to all other engines contained in the train set. The speed and direction controls of the Train Window always reflect the status of the selected individual vehicle rather than the complete train set. If there are several engines with different speed characteristics assigned to a train set (multiple unit), i.e. the engines run with different speed at the same speed step, then **TrainController™ 7** is able to balance out the different behaviour of the engines. This requires correct adjustment of the *speed profile* of each affected engine, however (see chapter 3).

Multiple unit operation via train sets is also possible with the throttle of your digital system. To operate the speed or direction of a train set select an arbitrary engine currently contained in this train set on the throttle of your digital system. Changing the speed or direction of this vehicle with the digital throttle will also be applied accordingly by **TrainController™ 7** to all other engines contained in the train set.

To operate the auxiliary functions of a certain engine or car select this particular vehicle in the Train Window, too. The operation of functions, however, only applies to the currently selected vehicle. The function buttons of the Train Window always reflect the status of the currently selected vehicle. In other words: for manual operation of functions it does not matter, if the vehicle is currently contained in a train set or not. The effect is limited to the particular vehicle.

Like in real railroads a certain vehicle can only run as part of one train at a time. If a vehicle is successfully added to a train set, then it is automatically removed from its previous train set, if any.

Cars and Load

For realistic simulation of train tonnage it is possible to specify the full weight and the empty weight for each car. With a specific menu command it is possible to toggle between both weights, i.e. it is possible to simulate loading and unloading of cars. The currently selected weight (load condition) of each car is applied to the calculation of the maximum speed or the acceleration momentum of affected train sets.

Cars can also be loaded or unloaded automatically during operation, e.g. during a running schedule.

Automatic loading and unloading of cars provide also the possibility to decrease the acceleration momentum of trains running in hidden areas of the layout, if desired. To accomplish this automatically, specify appropriate schedule operations, that unload all cars, when they enter the hidden area, and that load all cars again, when they leave these areas.

Forwarding of Functions in Train Sets

Automatic operation of auxiliary functions called for a train set are normally performed by the first engine or car only.

In order to allow auxiliary functions to be performed by other vehicles in a train set, too, it is possible, to turn on **function forwarding**. This is accomplished with a certain train operation, with which function forwarding can be turned on or off. This operation is usually used in macros. Assume a macro, with a first operation to turn function forwarding on and a second operation to call an auxiliary function to turn on the lights. If this macro is called for a train set, then the lights are turned on in all vehicles in the train set, which are able to perform this function.

The opposing operation, i.e. turning off function forwarding, is also available in order to return to the default policy.

Joining and Separation of Train Sets

Vehicles can be added and removed from train sets by using certain menu commands. Additionally **TrainController™ 7** is able to add or remove vehicles from train sets automatically during operation without explicit human intervention.

To accomplish this each train set can be separated into two parts at a time. This can be done manually by calling an according menu command or automatically. If a train is separated, then **TrainController™ 7** draws a red triangular marker between the two separated vehicles. It is also possible to join a separated train set.

If the speed of an engine contained in a separated train set is changed, then this change only applies to the engines contained in the same part of the separated train set. If, for example, a train set consisting of two locomotives is separated and the first engine is accelerated, then the second engine remains standing still. If the first engine is detected in an adjacent block, then the train set is finally dissolved and the second engine remains in its current block.

Note, that for reasons of simplicity only one separation is supported per train set. If a train set, which is separated between the second and third vehicle, for example, is additionally separated between the fifth and the sixth vehicle, then the red marker is moved from its previous position to the gap between the fifth and sixth car and the second and third car are joined again.

If a train set shall be divided into three or even more parts, then it has to be separated into two parts first. Before it is possible to separate one of these parts further it is necessary, that the other part leaves the current block first.

Arranging Train Sets by Train Tracking

If one part of a separated train set is manually moved to another block and detected there by train tracking, then the moved vehicles are automatically removed from the train set. The other vehicles form a new train, which remains in the previous block. The moved vehicles also form a new train, which is now located in the new block.

The opposing feature, i.e. joining trains by manual driving a locomotive or a train into a reserved block with other vehicles waiting there, is also possible. This is accomplished by checking the menu item **Enable Join by Train Tracking** in the **Train** menu for a

specific locomotive or car. If such locomotive is manually driven into a reserved block with other vehicles already located there, then the locomotive and the vehicles are automatically joined and form a new train set.

The joining process begins, when the approaching train enters the block, where another train is already located. In this moment both trains are displayed in the block with a red triangle between them. Actually both trains form a single train set now, which is still separated, however. The separated train set is finally joined and the red rectangle disappears, when the approaching train stops. From this moment all engines in the new train set are operated together.

The maneuver outlined above even works, when the approaching train is operated with the throttle of your digital system. After the join has been completed, all engines of the new train set can be driven with the throttle of the digital system by operating one of the engines with the throttle. The speed of the other engines is automatically synchronised by the computer software. In this way multiple units can be created very conveniently and without additional intervention by simply driving trains to positions, where other trains are already located.

Even though the menu item **Enable Join by Train Tracking** can be set for locomotives or cars only, it has also an affect for train sets. If a train set with a vehicle at the according end with this menu item checked enters a reserved block, then the train set and the vehicles located in this block are automatically joined and form a new train set, too.

In order to make the above maneuver to work properly it is necessary, that the train, which enters the block, can be detected in this block. For this reason it is necessary that the entering train can trigger an indicator in this block, even though there are already other vehicles located there.

Arranging Train Sets by Schedules

It is also possible to start a schedule with a separated train set. Depending on the direction of travel forced by the schedule, however, only the corresponding part of the separated train set will start to move. When this part of the train set enters the next block, the train set will be dissolved by leaving the non-moving vehicles in the start block of the schedule.

The opposing feature, i.e. joining trains at the end of a schedule, is also possible. This is accomplished with a specific schedule rule, that allows trains to enter destination blocks of schedules, that are reserved by other trains. If the train, which executes a schedule with this rule turned on, enters a destination block of this schedule reserved by another train, then both trains are automatically joined to form a new train set. This is the only

exception from the basic principle, that no train may enter a block, which is reserved by another train. This exception is also only available for destination blocks of schedules. In order to make this maneuver to work properly it is necessary, that the train, which runs the schedule and which shall join the other train already sitting in the destination block, can be detected in the destination block. For this reason it is necessary that the entering train can trigger a brake and a stop marker in the destination block, even though there is already another train located in this block.

Arranging Train Sets By Operations

Train sets can also be joined or separated by operations, macros and engine functions. The following actions are possible:

- join a separated train set
- separate the first locomotive located at a certain end of the train from the train set
- separate all locomotives located at a certain end of the train from the train set
- separate the vehicle (locomotive or car) located at a certain end of the train from the train set
- separate a train set at a certain side of a vehicle

The end of the train, that shall be separated from the train, is always specified based on the direction of travel relative to the layout (left vs. right; top vs. bottom). It is for example possible, to separate the first locomotive on the right end of the train. This is for example useful in a scenario, where an auxiliary locomotive is temporarily added to a train set to push a heavy train uphill. The direction of travel, which the block at the top of the grade is passed to by trains coming from the bottom, is always the same; e.g. from the right to the left. In such case the pushing locomotive is always located at the right end of the train., regardless whether the heading locomotive is running forward or backward. For this reason it is more useful to be able to separate the locomotive at the right end of the train rather than the locomotive at the tail of the train, which may point upwards, if the train set runs backwards.

In the above list the first four operations can be applied to train sets, usually as operations or macros executed at the beginning, at the end or during execution of a schedule.

The fifth operation can only be executed by vehicles (engines or cars), that are currently part of a train set. It is usually performed by a macro, which is called as an auxiliary function of the engine or car. This operation allows interesting maneuvers. Assume a train, that shall be separated left of the caboose, when it enters a certain block. If the operation to separate a train set at the left side of a vehicle is added to a macro and this macro is specified as an auxiliary function of the caboose, then each train set containing

the caboose can be separated left of the caboose by calling this auxiliary function; e.g. at the end of a schedule. Thereby it doesn't matter, at which position of the train set the caboose is placed.

Note, that the features described in this section, i.e. joining and separation of trains, are *logical* features. They are needed by the software to perform book keeping of the arrangements of train sets. At any rate it is also necessary, to care about the *physical* coupling or uncoupling of the particular vehicles contained in a train set. This is to be done by additional means not covered here.

List of enabled Trains and Train Groups

The usage of blocks, routes, schedules etc. can be limited to certain engines, cars and train sets. In this way it is possible to ensure, that certain schedules are only started with passenger trains or to avoid, that electric engines enter tracks without overhead cable.

If you want to define *home blocks* for certain trains, e.g. in a hidden yard, then include these trains in the list of enabled trains associated with these blocks. As a consequence only these trains will enter and stop in the affected blocks, while other trains will be automatically directed to other blocks.

Among others each block, route, schedule, brake or stop marker or turntable is associated with a list of enabled trains. Depending on the type of object these lists accomplish different purposes. In case of blocks, routes and schedules these lists control, which trains may use the particular block, route or schedule. In case of brake and stop markers the lists control, which trains the particular marker applies to. In case of turntables the list controls, which engines are turned to a certain direction.

Train Groups are useful in conjunction with these lists of enabled trains. In a train group several similar vehicles can be grouped together. Train groups rid you from entering the same set of trains repeatedly into different lists of enabled trains. If certain schedules shall only be executed by steam engines, for instance, then it is useful to create a train group, that contains all steam engines. Instead of adding the same set of steam engines to the list of trains of all affected schedules it is then possible to enter just this train group to these lists. If n schedules and x steam engines are affected, then the total number of list entries is reduced from n times x to n by using this train group.

In **TrainController™ 7 Gold** Train groups can be optionally defined to exclude all vehicles listed therein. Vehicles are contained in such train group, if they are not listed in this group.

The following rules apply to train groups and the list of trains associated with an object:

- An engine or car is contained in a train group, if it is a direct member of this train group, or if the train group contains another train group, that contains this vehicle.
- The list of enabled trains applies to an engine or car, if the vehicle itself is listed or if a train group is listed, that contains the vehicle.
- If the list does not contain any engines, then the list applies to all engines. This is in particular also true, if the list only contains only cars, but no engines.
- If the list does not contain any cars, then the list applies to all cars. This is also true, if the list only contains engines, but no cars.
- If the list applies to each particular engine or car contained in a train set, then the list also applies to the train set itself.
- As a consequence of the above an empty list, which is the initial setting of each object, applies to all engines, cars and train sets. By default and initially each block, route, schedule or other object may be used by all vehicles or train sets.

Examples:

- If a list of enabled trains shall apply to all steam engines, then create a train group of all steam engines and enter it into the list.
- If a list of enabled trains shall apply to all steam and diesel engines, then create a train group for all steam engines and a group for all diesel engines and enter both groups to the list.
- If a list of enabled trains shall apply to all freight trains, then create a train group of all freight cars (remember: create only one car for a set of freight cars, that is always running in the same formation) and enter it to the list. If there is no engine in the list, the list applies to all engines, that pull only freight cars.
- If the steam engines on your layout are divided into steam engines for main lines and steam engines for branch lines, then it may be useful to create a separate train group for each type and to combine both groups to the train group of all steam engines.
- If a list of enabled trains shall only apply to passenger trains pulled by steam engines on a branch line, then create a train group for the branch line passenger cars and a group for the branch line steam engines and assign both groups to the list.
- If you are in possession of only a few electrical engines but many steam and diesel engines and a list of enabled trains shall only apply to all non electrical locomotives, then create a list of the few electrical engines and specify the option, that all list entries shall be excluded from this group.
- If a list of enabled trains shall apply to engines only, that do not pull any cars, then a specific case applies. Create a train group, that contains the desired engines and another train group, that excludes all cars. Then enter both groups to the list of

trains. The group of desired engines can be omitted, if the list shall apply to all engines.

Operation of Additional Function Only Decoders

Function only decoders are often used to add additional functions to a decoder controlled locomotive or to other rolling stock. An example is lighting in passenger cars. These decoders can be controlled with **TrainController™**, too.

This is done by setting up a car with the digital address of the function only decoder. The function setup of this decoder is done as outlined in section 3.6, “Headlights, Steam and Whistle”.

Manual operation of the extra functions provided by the function only decoder is done by selecting the car in the Train Window and operating the function buttons of this car.

Example: Automatic Car Lighting

The following example demonstrates how a train can be prepared in order to operate the car lighting in this train automatically. It is assumed that the lighting is controlled by an additional function only decoder. Perform the following steps:

- Create and setup an engine for the actual engine heading the train.
- Create a car and specify the digital address of the function only decoder.
- Setup the function symbols for the functions provided by the function only decoder in the car. Use a unique function symbol for the car lighting, that has not been already used for functions of the actual engine.
- Arrange the engine and car in a train set.
- Assign the function symbol representing the car lighting to the operations of a *schedule*, a *macro* or an *indicator* (see also **Diagram 25**) as desired.

10.2 Acceleration and Train Tonnage

An additional feature of **TrainController™** is the realistic simulation of *momentum*, i.e. *acceleration* and *deceleration* of engines and train sets.

For each *engine* you can specify the *power* (see also Diagram 9). The power affects the acceleration of the engine. An engine with more power is able to accelerate faster. The acceleration is also affected by the *type* of the engine. Usually a electric engine is able

to accelerate faster than a steam engine with identical power. This fact is also taken into account when the acceleration is calculated.

Cars and train sets provide even more realistic simulation of momentum. It is namely possible to specify the *weight* of each car. The higher the total weight of all vehicles in a train set, the longer is the time needed to accelerate the train to a certain speed or to decelerate the train. The maximum speed of a train is also limited by the total weight of the train.

If several engines are running as a *multiple unit*, then the *power* of each engine is added to the total power of the multiple unit. Since the total power is higher than the individual power of each particular engine the multiple unit is able to accelerate faster and to run with a certain train tonnage at a higher maximum speed.

The time needed to accelerate or decelerate an engine or train set is additionally scaled and shortened using the scale factor of the *Railroad Clock*. If for example the scale factor of the clock is 10, then the calculated times are shortened to the tenth part. Even this shortening, however, results in timing, which is often found too slow. For this reason it is possible to adjust the *inertia* of each engine individually. In this way it is possible to accelerate or decelerate an engine without inertia or with the inertia of a real engine. Any setting between these extreme cases can be selected. It is also possible to adjust the inertia for acceleration and deceleration separately (see Diagram 9).

Do not be concerned if this sounds too complicated - especially in the beginning. For each engine which is created **TrainController™** assumes default settings for power, train tonnage and momentum. You are not required to set it. The default values result in a moderate behaviour for acceleration and deceleration which can be adjusted with the *inertia* as desired. The additional features discussed in this section are only needed if you want to simulate the behaviour of real trains.

10.4 Monitoring the Maintenance Interval

For each engine and car the elapsed operation time since the last *maintenance* is tracked by **TrainController™ 7 Gold**. This time is increased accordingly when a vehicle is running.

Based on the recommendations of the manufacturer of your engine or car, you can determine when it is time to lubricate the wheels or to change the carbon brushes. After maintenance you can reset the elapsed time to 0 (see Diagram 31).

For each vehicle it is possible to specify an individual maintenance interval and an optional operation, that shall be automatically performed, when the maintenance interval expires. The following operations are possible:

- Decommission of the vehicle.
- Display of a message in the Message Window.
- Execution of a macro.
- Execution of a schedule .

Especially the latter allows for very interesting features. It is for example possible to specify a certain schedule (*maintenance schedule*), which directs each vehicle automatically to a certain track of your layout, when the maintenance interval expires. If this feature is applied to a car, then the train set, which the car currently belongs to, is started by the maintenance schedule. The maintenance schedule should be equipped with an appropriate trial time. This ensures, that the schedule is also executed in cases, where the vehicle is currently busy in another schedule, when the maintenance interval expires. In this case the vehicle will first terminate its current regular schedules and then start the maintenance schedule.

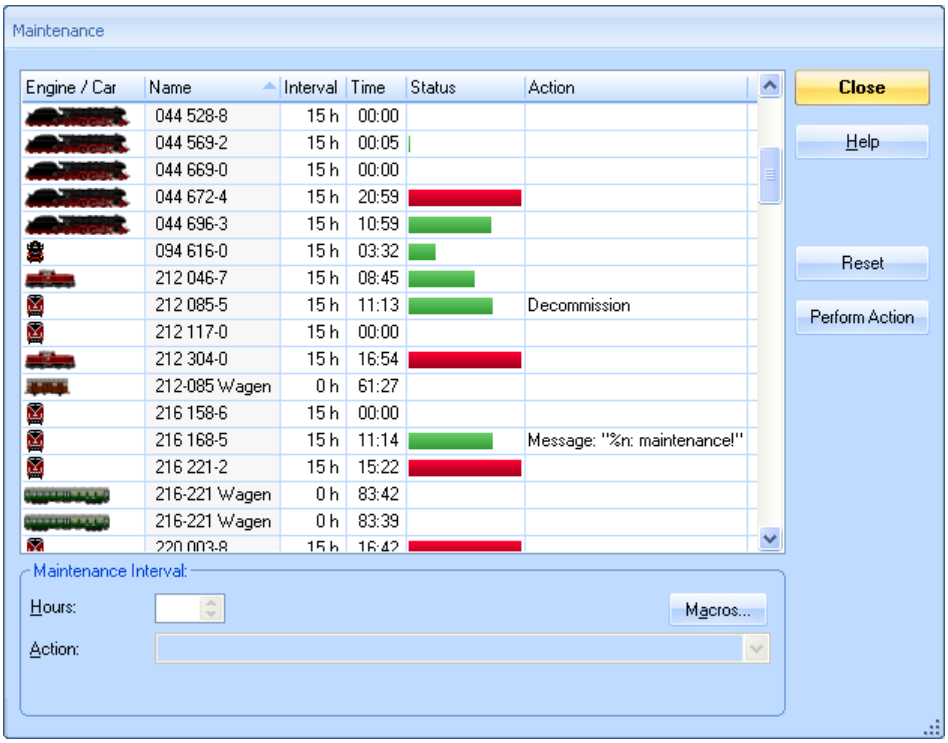


Diagram 31: Vehicle Maintenance

13 Indicators and Semi-Automatic Control

13.1 The Memory of Indicators

TrainController™ 7 provides an additional method to turn off the *indicator*:

- **With Indicator:** if this option is selected, then the indicator remains turned on until another indicator is turned off. Previous software versions only provided the possibility to leave the indicator turned on, until another indicator is turned on.

13.2 Protection and Locking with Conditions

Numerical Groups

In addition to the AND- and OR-groups provided by other **TrainController™** versions **TrainController™ 7 Gold** provides three additional types of groups:

- AT-LEAST-group: such group meets the condition, if at least a certain preset number of items contained in this group have the required state.
- AT-MOST-group: such group meets the condition, if at most a preset specified number of items contained in this group have the required state.
- EXACT-group: such group meets the condition, if exactly a certain preset number of items contained in this group have the required state.

These groups can be used to evaluate, if the number of items, which are in a required state, exceeds, falls below or fits a certain preset number. This option is useful, for example, to start a certain schedule, when at least three trains are waiting in a station, or to prevent trains from running to a hidden yard, if at least 5 trains are already stored there, etc.

Combined Groups

The COMBI-group provided by **TrainController™ 7 Gold** is another, very specific type of logical group, which provides interesting possibilities. A COMBI-group is a list of trains, blocks or schedules.

COMBI-groups can be used to check, whether certain trains are located in certain blocks and/or whether these trains are performing certain schedules. They can also be used to check, whether certain blocks are currently involved in certain schedules.

- A COMBI-group meets the condition, if at least one of the listed trains is currently located in at least one of the listed blocks and executes at least one of the listed schedules.
- If no block is listed, then the COMBI-group meets the condition, if at least one of the listed trains executes at least one of the listed schedules.
- If no schedule is listed, then the COMBI-group meets the condition, if at least one of the listed trains is currently located in at least one of the listed blocks.
- If no train is listed, then the COMBI-group meets the condition, if at least one of the listed blocks hosts a train, which executes at least one of the listed schedules.

This sounds complicated but is actually not very difficult. The following examples may help to understand the concept:

Examples:

- A COMBI-group, that contains the train “Freight Train”, the block “Mainline East” and the schedule “Local Freight”, meets the condition, if the “Freight Train” is located in block “Mainline East” and if this train is currently executing the schedule “Local Freight”.
- A COMBI-group, that contains the train “Big Boy” and the block “Northville Branch”, meets the condition, if the “Big Boy” is located in block “Northville Branch”.
- A COMBI-group, that contains the train “Passenger” and the schedule “Rhein**Gold**”, meets the condition, if the train “Passenger” is currently executing the schedule “Rhein**Gold**”.
- A COMBI-group, that contains the block “Southtown 1” and the schedule “Southtown - Northville”, meets the condition, if there is a train located in “Southtown 1”, which is currently executing the schedule “Southtown - Northville”.

The following features and limitations apply to COMBI-groups:

- Additionally to single trains it is also possible to include train groups in COMBI-groups. COMBI-groups can be included in other groups (such as AND-groups or OR-groups). COMBI-groups are the only type of groups, where trains and train groups can be included in. Adding trains or train groups to other groups (such as AND-groups or OR-groups) may cause unpredictable results. COMBI-groups must only contain entries referring to trains, train groups, blocks or schedules. Other entries, including other logical groups, contained in COMBI-groups are ignored.

13.3 Operations

System Operations

An additional feature is *system operations*. Among others the following system operations are available:

- Playing of sound files
- Execution of an external program
- Output of a warning tone with the speaker of the computer
- Turning off the power of the digital system
- Display of a message in the message window.
- Inserting a delay into the operational sequence.
- Start and stop of the clock

With these system operations, you are able to create, for example, an emergency stop button in your switchboard.

Train Operations

Train operations can be applied to trains. They are often executed by indicators, markers or schedules. They can also be executed by macros, which again are executed by schedules or as a train function.

Among others the following train operations are available:

- Execution of a train function
- Stop a train with or without momentum.
- Set the train direction.
- Start a train with interlocking

- Termination of the schedule, which is currently executed by the train
- Set a temporary speed limit.
- Turn on or off function forwarding.
- Join or separate train sets.
- Automatic load or unload of cars.

13.7 Decommissioning of Objects

X

Blocks, routes, schedules, trains, turnouts and other objects can be decommissioned at any time during operation. Decommissioned objects are excluded from operation. Decommissioned schedules, for example, cannot be started; decommissioned routes cannot be activated; decommissioned trains cannot be driven. A decommissioned turnout cannot change its state; such turnout can still be used in routes, however, if the current state of the turnout matches the turnout state required by the route.

Decommissioning is useful for each object, that shall be prevented from being operated during operation of the layout. An object can be decommissioned at any time during operation. It is not necessary, to turn on edit mode before.

Objects are decommissioned by clicking to the object with the right mouse button and then selecting the command **Decommission** in the popup menu. If this command is applied to a decommissioned object, then the object is put into operation again.

Even though decommissioned objects cannot be operated, they can still be accessed and selected in lists or referred to by other elements without limitation. In particular decommissioned objects do still exist.

13.8 Turnout Position Control

X

Turnout position control can be used to protect turnouts, that are currently locked in routes, against outside interferences or operation failures.

Turnout position control is based on different categories of turnout status:

- The digital system stores and reports back the most recent turnout command (logical turnout state). This information can be for example used to detect, if a turnout is operated by an external handheld.
- The turnout decoder can report back the electrical status of the turnout drive. This feature usually requires a turnout decoder, which is able to report back the current

status to the digital system, and certain circuitry associated with the turnout drive, which reports the turnout status back to the decoder. This information can be used to detect, if the turnout drive failed to execute a turnout command issued by the digital system.

- The electrical status of the turnout is reported back to the digital system or the computer, respectively, by feedback input contacts, which are associated with a separate feedback encoder. This information can be used, too, to detect, if the turnout drive failed to execute a turnout command issued by the digital system.

TrainController™ 7 Gold provides turnout position control, too. For this purpose it supports all methods (a) to (c) listed above.

Turnout position control is usually only relevant for turnouts currently locked in routes. Since routes are just the tool to assure certain turnout positions, it should not matter, if a turnout changes its position, if it is not locked by a route. For this reason turnout position control in **TrainController™** only applies to turnouts locked in routes, too.

If turnout control is turned on for a specific turnout, then **TrainController™** automatically evaluates the turnout status reported back by the digital system (method (a) and (b)) to determine, whether the turnout position is in line with the associated route, if any.

If the digital system is not able to report back to the computer the status of turnouts, or in certain other situations it may be useful , to add method (c). To support this method **TrainController™ 7 Gold** allows to assign an individual feedback address and feedback status (on or off) to each particular position of a turnout. The feedback status is then additionally evaluated to determine, whether the turnout position is in line with the associated route or not.

By specifying a certain delay for each turnout **TrainController™** can be caused to check, whether a turnout requested by a route is in line with the route, when the specified delay has passed. The route is only activated, when the delays of all turnouts in the route have passed and if each turnout is in line with the route.

Additionally to the delayed check of turnout positions prior to final activation of each route **TrainController™** also reacts to turnout or feedback reports, which indicate, that a certain turnout has changed its position and that it is no longer in line with the route, that currently locks the turnout.

Error Processing

Turnout position control does not make any sense, if there is no reaction to failing turnouts.

One of the following mandatory reactions must be individually selected for each turnout. The selected reaction is performed, when the turnout is locked by a route for a train in a schedule:

- Search alternate path: if this option is selected, then **TrainController™ 7** tries to continue the affected schedule with an alternate path.
- Lock block exit: if this option is selected, then **TrainController™ 7** locks the exit of the block, where the affected train is currently located. This will cause the train to stop in this block and enables the human operator to clear the turnout problem.
- Stop schedule: if this option is selected, then **TrainController™ 7** terminates the affected schedule. This is another, more drastic measure to prevent the train from passing the failing turnout and to enable the human operator to resolve the problem first.

Additionally and optionally it is possible to decommission the failing turnout in its current position and/or to execute a macro to perform other actions. Note, that decommission of the turnout does not prevent the turnout from being used by routes, that match the current position of the turnout. If a turnout, that failed to go to the thrown position, for example, is decommissioned, then it can be still requested by other routes in the closed position.

Limits of Turnout Position Control

While the methods (a) to (c) listed above concern the logical or electrical status of a turnout, the actual physical/mechanical position of the turnout, i.e. the actual position of the switch blade, can differ from the electrical state. This is for example the case, if the turnout drive operated the turnout correctly, but a small piece of ballast prevents the blade from following the drive completely. Such mechanical problems usually remain undetected or require at least complex and uneconomical changes of the turnout construction, which enable the blade to report back its position to a feedback input according to method (c). For this reason turnout position control can usually only be used to solve problems related to the logical or electrical status of turnouts, e.g. unauthorized operation of locked turnouts by external handhelds or electrical problems in conjunction with the decoder or turnout drive. Turnout position control can usually not solve mechanical turnout problems.

Because of these undetected mechanical problems and because the error processing of failing turnouts detected by turnout position control always causes an actual unwanted intervention into the normal operation of the layout, all measures to prevent turnouts from failing should be utilized first. Turnout position control is a measure, that can be added as an additional security measure for normally reliably operating turnouts. It should not be misunderstood as a compensation for unreliably working turnouts!

14 The Visual Dispatcher II

14.1 Manually created Block Diagrams

In section 5.2, “Blocks”, you have been made familiar with *block diagrams*. These diagrams contain *block* and *routes* between blocks and describe the track layout of your entire model railroad in rough outline.

Normally each block diagram is associated with a switchboard and automatically created by the software by using the information contained in this switchboard. It is also possible to draw additional block diagrams by yourself and to create custom block diagrams. In Part I of this Users Guide it was always assumed, that block diagrams are automatically calculated by the software. In rare situations, however, it may be necessary, to extend these automatically created diagrams by self-drawn, custom block diagrams. Custom block diagrams are required, if a part of the layout shall be integrated into the block system of the dispatcher, which cannot be represented in a switchboard track diagram very well. One example is the use of the Müt Lok-Lift.

It is also possible to extend an automatically created block diagram by own objects by turning the calculated block diagram into a custom block diagram.

TrainController™ 7 offers the following features to manage your block diagrams:

- Creation of a calculated block diagram for each switchboard.
- Deletion of not needed or empty block diagrams. An empty block diagram does not contain any blocks. Empty block diagrams can be generated, if a switchboard does not contain any block symbols.
- Turning a calculated block diagram into a custom diagram to allow extension with self-drawn objects.
- Turning a custom block diagram back into a calculated block diagram, if the diagram was originally created as a calculated block diagram for a certain switchboard.

These features should only be used in exceptional cases and by experienced users, because they may cause serious impacts on your layout data.

Routes

Routes are used to connect blocks with each other. If there is a track connection on your layout, on which trains can travel from one block to another, then a route must be drawn between both blocks. A route represents a track connection between two blocks. Routes are displayed by lines.

The following image explains the terms once more:

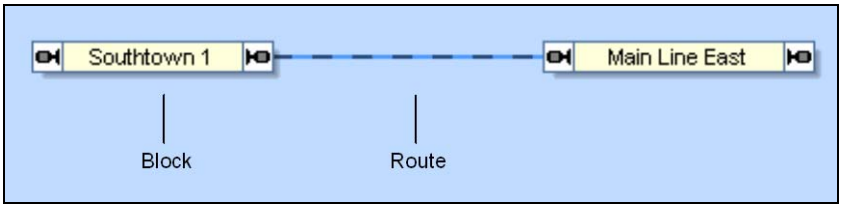


Diagram 32: Blocks and Route

In the diagram displayed above the blocks “Southtown 1” and “Main Line East” are connected with a route.

Please pay attention that the blocks and routes are arranged accordingly. A certain block can only be passed without stop when the train can enter the block through one entry and leave the block through the opposite exit.

Diagram 33: Passing Blocks and Routes without Stop

In the diagram displayed above a train can pass block “Southtown 1” without stopping and reversing its direction. Coming from block “Main Line West”, for example, a train will enter “Southtown 1” through the left entry and can leave this block through the opposite exit to proceed to block “Main Line East”.

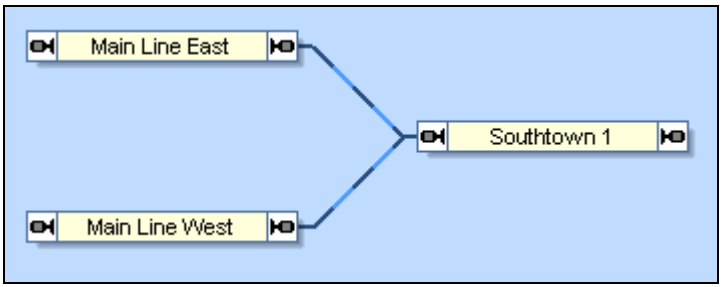


Diagram 34: Blocks with enforced change of Direction

This arrangement allows also for train movements from “Main Line West” via “Southtown 1” to “Main Line East”. But after arriving in “Southtown 1” a train has to stop and to reverse its direction in order to proceed to “Main Line East”, because it enters and leaves the block on the same side.

Even though you are not forced to do so it is recommended to draw the block diagram according to the actual conditions of your layout. If you follow the layout of your model railroad you will probably not encounter any problems.

Nevertheless you should always check whether the routes between your blocks touch the blocks at the correct side.

Please note also that a train cannot run from one block to another under control of the Dispatcher if there is no route between both blocks.

It is not necessary to draw routes always as straight lines. If desired for display purposes these lines can also contain corners. Such corners are only used for clearer visualization of the diagram, they do not have any impact to the operation of trains.

Guidelines for routes:

- Routes must touch the blocks at the correct entries/exits, because this affects the direction of travel through the related blocks assumed by the *Dispatcher*.
- Routes are to be used, if the track connection between both blocks contains turnouts or crossing.
- Each pair of blocks can be connected by an arbitrary number of routes.

14.4 Controlling the traffic flow in Schedules

Train Guidance based on Train Length

The length of locomotives, cars and train sets can affect, how schedules are executed.

For this purpose it is possible to specify a maximum train length for each block. This maximum train length describes, up to which length trains fit into the particular block. Together with the length specified for each train and car or calculated for each train set, respectively, **TrainController™ 7** can determine, whether a train fits into a particular block or not.

This is used to accomplish the following goals:

- Trains can be prevented from driving to destination blocks of schedules, which are shorter than the length of the train.
- Trains can be prevented from performing unscheduled stops in blocks, which are shorter than the length of the train. If such blocks are reserved by longer trains, then these blocks will be treated in a similar way like critical blocks.
- Trains can be prompted to prefer the shortest destination block, which is long enough to store the train.

The features listed above can be activated by specifying a maximum train length for the concerning blocks and by checking certain schedule rules.

These features are in particular very useful in conjunction with train sets, that change their formation during operation. If such changes also affect the length of the train set, then this may change the destination, to which trains run or where trains perform unscheduled stops. If trains entering a hidden yard always prefer to drive to the shortest destination block in this yard, which is longer then the train itself, it is possible to utilize the available track space optimally. In such case no train will waste track space by run-

ning into a track, that is longer than necessary. Since **TrainController™ 7 Gold** is able to calculate the length of train sets, the track space in the hidden yard can be even used optimally for train sets, which change their formation during operation.

The following aspects should be also noted:

- The maximum train length specified for each block does not have any effects to the braking ramp of trains or to the location where trains stop in the block. There is no correlation between the maximum train length of a block and the distances specified for shifted brake and stop markers (see page 63).
- If a zero length is specified for a vehicle or all vehicles in a train set, respectively, then this train fits into all blocks. Zero length is the initial default setting for each vehicle.
- If a zero length is specified for a block, then all trains fit into this block. The block is assumed to have an "unlimited length". Zero length is the initial default setting for each block.

Forcing a Train to start a Schedule in a certain Direction

Normally each train, which is suited to execute a certain schedule, can be started in both directions, i.e. forward or backward. Therefore it does not matter, with which orientation the train is standing on the track. It will be started in the right direction of travel in each case.

With specific schedule rules, however, it is possible to force all trains started by the concerning schedule to start in a certain direction, i.e. forward, backward or by maintaining their current direction.

If the rule to start in forward direction is activated, for example, then a train will not be started by the concerning schedule, if it had to run in backward direction. The above is correct for single locomotives. For train sets the rule works slightly different. In conjunction with train sets 'forward' is interpreted as 'pulling' and 'backward' means 'pushing'. If the rule to start in forward direction is activated, for example, then a train set will only be started by the concerning schedule, if a locomotive is located at the end of the train, which corresponds to the requested direction of travel. In other words: the train set will only start, if there is a locomotive, which will pull the train set. If the rule to start in backward direction is activated, then a train set will only start, if there is a locomotive, which will push the train set. Care has to be taken, if there are locomotives located at both ends of a train set. In this case the train will be pulled and pushed, regardless in which direction it is started. In this case the rule does not have any affect

and will not prevent this train set from being started, regardless of the requested direction of travel.

Routes with separate occupancy indication

Release of routes can be controlled individually and independently from the occupancy state of adjacent blocks. It is possible to assign a set of indicators to each turnout or route. These indicators determine whether a route is occupied or not. If at least one of these indicators is turned on, then the route is assumed to be occupied. It is possible to assign the same indicator to more than one turnout or route.

- A route is assumed to be occupied, if at least one of the indicators is turned on, which are assigned to the route.
- A route is also assumed to be occupied, if it contains one or more turnouts and at least one of the indicators is turned on, which are assigned to these turnouts.

It does not matter, whether a certain indicator is directly assigned to a route or assigned to a turnout contained in this route (indirect assignment). The assignment of indicators to turnouts is more convenient in cases, where many routes pass the same common turnout. To accomplish occupancy indication for all these routes in one step it is sufficient to assign the indicator to the common turnout. Assignment of indicators to routes, on the other hand, is useful for routes, that do not contain turnouts at all, or if the occupancy indication of the route depends on indicators, that cannot by associated with turnouts.

Indicators associated with turnouts and routes, respectively, should be preferably created in **TrainController™ 7** as part of the properties of each turnout or route, respectively, rather than as separate switchboard symbols.

Occupancy indication of routes allow routes to be released independently from the occupancy state of adjacent blocks. Blocks or routes are usually not released, until the train reaches a stop indicator in a subsequent block. If your routes are equipped with an own occupancy indication, it is possible to turn off this rule. In this case routes can be already released, when the train reaches the first indicator at the entrance of a subsequent block, assumed, that the route is not reported as occupied anymore. In this case the track area covered by such routes is earlier available for other trains.

The rule to release preceding blocks and routes of a certain schedule at the stop indicator of subsequent blocks should be turned off only, if the routes contained in this schedule are equipped with an own occupancy indication. Furthermore the tail of each train should be able to trigger the sensors used for occupancy indication of the routes. Usual-

ly this requires, that the trailing cars of affected trains are lighted or the wheel sets of these cars are conductive.

Occupancy indication of routes can also be used to detect cars lost in a turnout area or to prevent routes from being prematurely released, when a long train is completely filling the subsequent block, but the tail of the train is still located on the route. In this case the route is not released, even if this long train has reached the stop indicator of this block (exception: the block is the destination block of a schedule, in which case all routes contained in this schedule may be released).

Schedule Watchdog and Limited Aberration Protection

With a specific schedule rule it is possible to specify a schedule watchdog. This is the maximum time period between activation of two indicators. If no indicator is triggered within the specified period of time and the train is set to run at non zero speed, then it is assumed, that the train got stuck. In such cases a warning is displayed in the message window and an error indication is raised in the train list.

It is furthermore possible to activate a limited aberration protection for each schedule. If the train under control of this schedule is detected by train tracking in block, which does not belong to the schedule, then the train is stopped immediately, a warning is displayed in the message window and an error indication is raised in the train list. Such condition can occur, for example, if a turnout contained in the schedule does not work correctly and the train is directed to a wrong block.

The limited aberration protection can detect certain, but not all possible conditions, where trains run into wrong blocks. In particular it usually cannot protect the train against collision, if the wrong block is already reserved by another train. Aberration protection should not be misunderstood as collision protection and it does not rid you from arranging for reliably working hardware.

Track Cleaning Trains

With a specific schedule rule it is possible to specify, that always that path is selected, that contains routes or blocks, which have been visited by the train under control of this schedule the longest time ago (“oldest” block or routes).

If two or more identical paths in a schedule are available, then the selection is usually performed by random. By using the option to select the “oldest” block or route the schedule will be performed in a more regular or systematic manner. This option ensures, that a train under control of this schedule selects another path each time it passes

a certain branching. This option is only effective, however, if the available paths are identical and no obstacle is blocking a certain path. The option is also only effective on a per schedule and per train base. This means in particular, that all time stamps are cleared, when the schedule is terminated. Thus this option is only useful, if the train passes the same branching in the same schedule run several times; for example in cycle or shuttle schedules.

This option can also be used to arrange automatic schedules for track cleaning trains. Due to the factor, that the train will prefer to go to that blocks or routes, that have been visited by this train the longest time ago, it will sooner or later visit all blocks and routes it can reach in this schedule, assumed the schedule is specified as cycle or shuttle with an appropriate number of repetitions. This is a good qualification for complete and systematic track cleaning.

14.6 Timetables

The timetable, which is displayed as a part of the dispatcher window in **TrainController™**, is contained in a separate *Timetable window* in **TrainController™ 7**.

15 Turntables and Transfer Tables

15.1 Introduction

Turntables and *transfer tables* are used in **TrainController™** to operate real turntables and transfer tables on your model railroad with the computer. In this document the term "turntable" is mostly used synonymously for both, turntables and transfer tables.

TrainController™ provides a separate turntable window, which provides a graphical representation of each turntable or transfer table and which allows manual operation of turntables.

Different turntable windows can be opened simultaneously to control several turntables/transfer tables at the same time. The number of turntable windows is only limited by the capacity of your computer.

Each turntable object can be configured to operate a turntable or to operate a transfer table as displayed below:

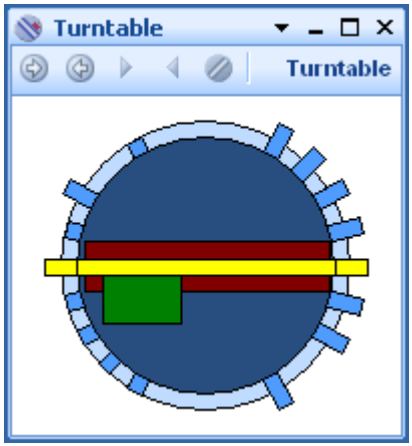


Diagram 35: Turntable Window

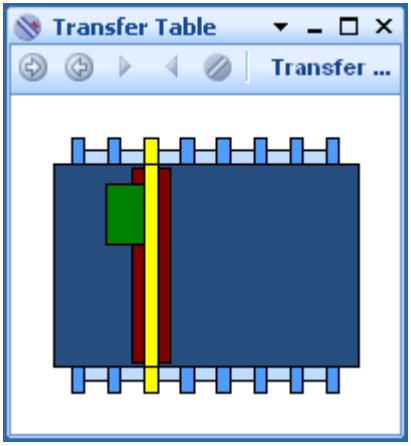


Diagram 36: Transfer Table

Special features are:

- up to 80 tracks on each turntable or transfer table
- each track can be individually configured as active or inactive as well as removed completely
- each turntable can be operated manually via the turntable window
- predefined software drivers for all leading turntable types
- generic turntables and transfer tables allow adaptation to custom driven devices
- each turntable/transfer table can be operated semi-automatically by operations of push buttons, macros, indicators or routes
- In **TrainController™ 7 Gold** the operation of turntables and transfer tables can be easily integrated into schedules, **AutoTrain™** or running of trains under interlocking

Supported Turntable/Transfer Table Commands

TrainController™ supports the following turntable/transfer table commands:

- permanent move in either direction
- stop of permanent move with automatic alignment to the next active track
- step to the next or previous active track
- direct selection of specific tracks *(indexing)*
- 180° turn (turntables only)
- dedicated adjustment of locomotive direction during automatic operation (turntables only, see page 123)

Integrating Turntables into the Operation of the Layout

In **TrainController™ 7 Gold** turntables are created by inserting a turntable symbol at an appropriate location in a switchboard. This symbol allows to operate the turntable with the mouse via the switchboard. It is also optionally possible to open one or more *turntable windows* via the **Window** menu of **TrainController™**. The turntable symbol in the switchboard is optimised for space-saving display. For this reason inactive tracks, i.e. tracks without connection to the layout, are not displayed by turntable symbols in the switchboard. Because the symbol has furthermore to fit into the arrangement of switchboard cells, the layout of turntable symbols is compulsorily a bit schematic. The display in the turntable window on the other hand is more realistic. Both views can be used alternatively or simultaneously for manual operation and control of turntables.

The turntable symbol in the switchboard provides some advantages, however, that are not provided by the stand-alone turntable window:

- the turntable symbol integrates more smoothly into manual switchboard operation than manual operation via a separate window
- the turntable symbol visualises the linkage to adjacent blocks
- the turntable symbol supports simple integration of the turntable into automatic operation of the layout, because turntable symbols are taken into account by the automatic calculation of the block diagram (see section 5.2, “Blocks”). All possible paths from the bridge to adjacent blocks or back are automatically captured as routes
- turntable symbols are also visible in the associated calculated block diagrams
- since turntable symbols are usually associated with blocks, they can also display, which train is currently located on the bridge and how the train is oriented.

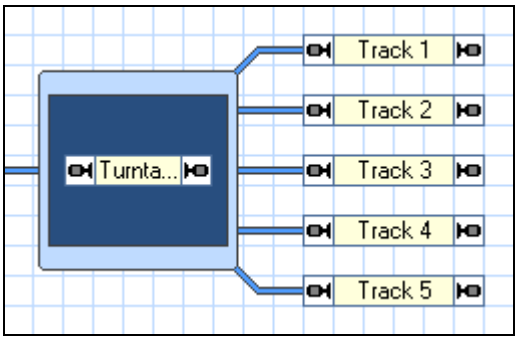


Diagram 37: Turntable Symbol in the Switchboard

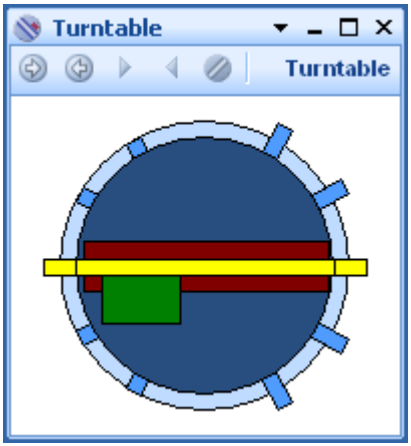


Diagram 38: Corresponding Turntable Window

Automatic Operation

Turntables and transfer tables can be easily integrated into automatic operation by using turntable symbols in switchboards. These symbols are automatically associated with a block and taken into account by the automatic calculation of the block diagram (see section 5.2, “Blocks”). All possible paths from the bridge to adjacent blocks or back are automatically captured as routes.

Right after configuring a turntable symbol accordingly in a switchboard it is possible to run trains automatically over the turntable. The routes, which connect the bridge (more

precisely: the block associated with the turntable) with adjacent blocks, can be used by AutoTrain or in other schedules like any other route, too. Actually with regard to automatic operation there is no fundamental difference between turntables and turnouts.

For each track it is possible to specify, that certain locomotives may leave the bridge via this track only with a certain orientation (forward or backward). In this way it is for example possible to force steam locomotives to enter the roundhouse only with a certain orientation, while Diesel or electrical locomotives may still enter the roundhouse with an arbitrary orientation.

Please note, that the turntable/transfer table must support indexing in order to be operated automatically. If you are using an analog or generic turntable/transfer table, then setup this turntable for indexing according to section 15.7, "Turntable Operations". Digital turntables support indexing and no further measures are necessary.

15.4 The Track Layout of a Turntable/Transfer Table

Active and Passive Tracks of Turntables

Each physical turntable or turntable decoder, respectively, can support a maximum number of track exits or tracks. The maximum number of tracks of the Maerklin digital turntable 7686, for instance, is 48. Usually only a fraction of the possible tracks are actually used.

The used tracks are divided into active tracks and passive tracks.

Active tracks correspond to those track exits of the turntable, which are connected to existing tracks of the layout. Engines can enter and leave the turntable via active tracks.

Passive tracks correspond to those track exits of the turntable, where the bridge of the turntable can be turned to, but which are not connected to existing tracks of the layout. In many cases there is only a short stub track associated with a passive track. Engines cannot enter and leave the turntable via passive tracks.

Diagram 38, for instance, shows a turntable with 6 active and 4 passive tracks. The total number of active and passive tracks must be always even.

Note, that all active and passive tracks are usually important in conjunction with physical control of the turntable bridge and the turntable decoder. The decoder does not care, whether an engine can leave the bridge via a certain track exit or not. For this reason the difference between active and passive tracks is irrelevant for the decoder. But the bridge must be able to turn the house to each existing track exit, regardless whether the exit is passive or not. In Diagram 38, for instance, there are 10 track positions, where the house of the bridge can go and thus all 10 positions, i.e. the number of active and passive tracks, must be programmed into the decoder, if any, as different positions.

Synchronizing the Turntable Symbol

The turntable symbol in the switchboard of **TrainController™ 7 Gold** only displays active track exits. In the switchboard it is important to save place and to visualise, how the turntable tracks are connected to the circumjacent track layout. For this reason the passive tracks, which do not have a connection to the track layout, are not displayed by the turntable symbol in the switchboard.

In order to work properly the turntable symbol in the switchboard must be synchronised with the track layout of the physical turntable.

Diagram 39 illustrates, how this is done:

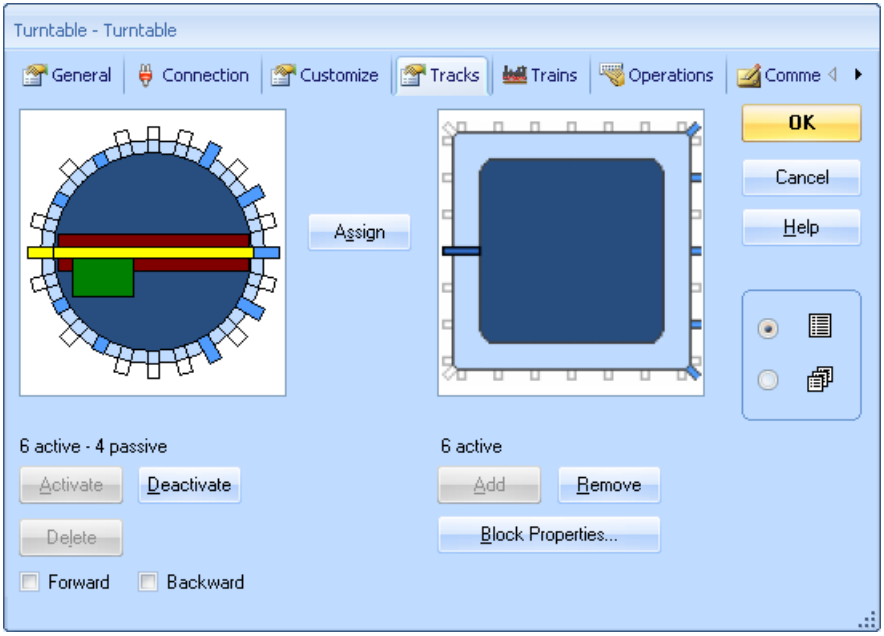


Diagram 39: Synchronizing the Turntable Symbol

The left image in Diagram 39 represents the track layout of the physical turntable. It has 6 active and 4 passive tracks, 10 significant track positions in total. The right image shows the schematic track layout of the turntable symbol in the switchboard. The number of active tracks must be identical in both images. Passive track exits of the physical turntable, that can be addressed with the bridge or the turntable decoder, respectively, but do not have a track connection to the rest of the layout, are not displayed and not taken into account by the schematic turntable symbol displayed in the switchboard. This is done to reduce the switchboard space required to display the turntable symbol.

To perform the synchronisation ensure first, that the number of active tracks in both windows is identical. Then select a track in the left image and a track in the right symbol, that shall be mapped to each other. Then press **Assign**. The subsequent procedure automatically iterates clockwise through the active tracks of both symbols and automatically maps the tracks of the physical turntable displayed in the left image to appropriate tracks of the symbol to the right.

Forward and Backward Tracks of Turntables

Each active track of the physical turntable, i.e. those tracks, that are actually connected to the layout, can be marked as a *forward track* or *backward track.*

These markings are taken into account during automatic operation of a turntable and do not apply to transfer tables. If a track is marked as forward track, then all affected locomotives, that leave the turntable bridge via this track during automatic operation, are automatically turned with their head to this track, so that they leave the bridge in forward direction. If a track is marked as backward track, then locomotives are automatically turned with their rear to this track, i.e. they will leave the bridge via this track in backward direction.

It is furthermore possible to specify, which locomotives are affected by these markings. This is done by filling the associated list of enabled trains (see also page 95) accordingly. In this way it is for example possible to force steam locomotives to enter the roundhouse only with a certain orientation, while Diesel or electrical locomotives may still enter the roundhouse with an arbitrary orientation.

Turning Locomotives automatically to an individual Direction

If the turntable is operated manually, then the provided commands provide full control over the direction, in which locomotives are turned.

For automatic operation it is possible to mark each active track of the turntable as forward or backward track as outlined in the previous section. This setting is usually valid for all trains, that exit the turntable via the tracks, which are marked in such a way. This feature is useful for tracks, that shall always be passed in a certain direction, e.g. if certain locomotives shall enter a roundhouse always in forward direction. In particular these settings apply to all schedules, that contain the turntable, in the same way.

Tracks, which are not marked as forward or backward direction are usually accessed by the turntable bridge on the shortest possible way. For such tracks the direction, in which the locomotive leaves the turntable, cannot be predicted. It is sometimes desirable, however, to have control over the direction as the case arises. For this reason there is an additional option, with which the direction, in which locomotives leave the turntable, can be set on a per schedule basis. In this way trains, that perform a certain schedule, can be caused to leave the turntable in forward direction; and trains, that perform other schedules, can be caused to leave the turntable via the same tracks in backward direction.

The following priority scheme applies for all locomotives, that pass a turntable under control of a schedule:

- If a certain direction for the exit of the turntable is set in the schedule specific settings of the block, that belongs to the turntable, then all locomotives under control of this schedule leave the turntable in the specified direction. This setting applies to all tracks of the turntable.
- if the above does not apply or the locomotive is controlled by AutoTrain or interlocking, then the locomotive leaves the turntable in the direction specified for the according turntable track.
- If no direction is specified neither for the schedule nor for the exit track, then this track is accessed by the turntable bridge on the shortest possible way. In this case it cannot be predicted, whether the locomotive leaves the bridge in forward or backward direction.

Appendix

Migrating Existing Data Files from TrainController™ 5

Text Elements

Text elements in **TrainController™ 7** are much more powerful than text elements in **TrainController™ 5**. The new text elements provide much more control of the position, size, alignment and displayed colours. These improvements required some incompatible changes, however. For this reason it cannot be granted, that all existing text elements are automatically converted into the new format without any perceptible deviations. This applies in particular to non-horizontal text elements. For this reason manual rework may be necessary for certain text elements to get them displayed in a similar way as in previous versions.

Traffic Boxes

In order to simplify matters the terms “block” or “block symbol” are used synonymously in **TrainController™ 7** for the term “traffic box” known from previous versions.

Route Symbols in Block Diagrams

Route symbols in block diagrams were already removed in Version 5.5 of **TrainController™**. They were transitionally still supported in Version 5.5 and 5.8 for compatibility reasons. In **TrainController™ 7** these symbols have been finally removed. All route symbols are now automatically turned into connecting lines without symbols, when a data file created with **TrainController™ 5** is being loaded. If a single connection between two blocks in an existing block diagram contains more than one consecutive route symbol or if a certain route symbol is only connected to one block, then the according route is deleted.

Assignments of Indicators to Routes

In **TrainController™ 7** indicators are always explicitly assigned to routes or contained turnouts, if required (see page 111). The sometimes confusing so called “Auto-Detect”-feature of **TrainController™ 5** has been superseded in **TrainController™ 7** by the

possibility to assign indicators to turnouts. Indicators associated with turnouts and routes, respectively, should be preferably created in **TrainController™ 7** as part of the properties of each turnout or route, respectively, rather than as separate switchboard symbols.

When a data file created with **TrainController™ 5** is being loaded, then all indicators implicitly associated with a route by the former "Auto-Detect" feature are automatically explicitly assigned to this route. This change does not affect the operation of your layout.

Entry Locks

Entry locks of blocks are directional in **TrainController™ 7.** If a data file created with **TrainController™ 5**, where such locks always affected both directions of travel, contains operations, which lock the reservation of blocks, then these operations are duplicated to directional locks for both directions of travel through each block.

Schedules in Version 4 Format

Schedules, which were created in Version 3 or Version 4 of **TrainController™** by using the at that time valid mechanism of *lines*, were imported into **TrainController™ 5** in a specific format. In particular the diagrams of those schedules were not based on the main block diagram. These specific schedules were transitionally still supported in **TrainController™ 5** for compatibility reasons. However, it was always recommended for several years now to convert the old format of these schedules into the new format introduced with Version 5.

These old-format schedules are not supported anymore. Affected old-format schedules are deleted during loading of a layout file created with an earlier Version of **TrainController™**.

Note once more, that only schedules created with Version 3 or 4 are affected by this change. Schedules created with **TrainController™ 5** are not affected and work as before.

End blocks of Schedules

TrainController™ 5 distinguishes between (dead) end blocks and destination blocks. By enabling a certain rule it is possible to use end blocks as destination blocks of schedules, too. This is no longer supported for schedules newly created in **TrainControl-**

ler™ 7. For reasons of clarity all desired destination blocks of schedules are always marked explicitly.

Schedules, that are contained in files created with **TrainController™ 5** may continue using end blocks as destination blocks for the time being. The according rule in the properties of the schedule should be turned off, however, at the next opportunity and the according destination blocks should be marked explicitly.

Migrating Turntables and Transfer Tables to TrainController™ 7 Gold

In **TrainController™ 7 Gold** turntables and transfer tables can now be easily integrated into the operation of switchboards and automatic calculation of the block diagram. This is taken into account during conversion of existing layout files by creating an additional switchboard containing a single turntable symbol for each existing turntable or transfer table. You should take advantage of these new possibilities by moving this symbol to an appropriate location of your existing switchboards. Routes, that you may have created in previous versions to involve your turntable into automatic operation, are still working. They should be deleted, however. The necessary routes are now automatically created by the software during the automatic calculation of the block diagram for the switchboard, where the turntable symbol is finally located. Do not forget, however, to assign the new routes to those existing schedules, which contain your turntable.

In **TrainController™ 5** it was possible to assign feedback indicators to the bridge and the tracks of each turntable or transfer table. This is no longer necessary in **TrainController™ 7 Gold**. Since the turntable symbol in the switchboard or block diagram shows the status of the block, which is associated with the bridge, the new display is much more informative now, because it shows also the train, which is currently located on the bridge. Assignment of occupancy indication to the particular tracks is no longer actually needed, too, because this information can be gathered from the display of the adjacent blocks in the switchboard or block diagram. For this reason it is no longer possible to assign feedback indicators to turntables or transfer tables newly created in **TrainController™ 7**. Existing assignments, however, can be still edited via the Turntable dialog for the time being.

The turn boundary, a difficult to understand feature of **TrainController™ 5**, has been superseded by the much more plain approach of forward and backward tracks. Most objects, that control a turnout via their operations and by using the old scheme of left and right orientation, such as existing routes, are automatically converted to new operations based on forward and backward operations. Some very specific and extremely rarely used operations of previous versions, namely turning the bridge by inverting the train

orientation and turning the bridge by maintaining the train orientation are not supported anymore and converted to moves of the bridge to the destination track on the direct (shortest) way.

Migrating Turntables and Transfer Tables to TrainController™ 7 Silver

This section only applies to **TrainController™ 7 Silver**. In **TrainController™ 5** it was possible to integrate turntables and transfer tables into automatic operation by means of operations, that are executed by other objects (e.g. routes). With these operations it was not only possible to move the bridge to a specific track, but also to turn the locomotive automatically into a specific direction (e.g. heading to the right when leaving the bridge). **TrainController™ 7 Silver**, however, yet provides operations to move the bridge to a certain track but does not support anymore automatic turning of locomotives to specific directions. The new turntable operation "go to track with marker (house)", however, may provide an adequate compensation. With this operation it is at least possible to turn a locomotive to a specific direction, when the direction of the locomitve is known in advance. The operations in existing layout files, which turned a locomotive to a specific direction, are converted accordingly. That means: after conversion to the format of **TrainController™ 7 Silver** these operations will still move the bridge to the same tracks as before, but they won't care about the resulting direction of the locomotive.

The turn boundary, a difficult to understand feature of **TrainController™ 5**, was needed in previous versions to be able to turn a locomotive to a certain direction (e.g. heading to the right when leaving the bridge) during automatic operation. Since this is no longer supported by **TrainController™ 7 Silver** the turn boundary became obsolete and is no longer available.

In **TrainController™ 5** it was possible to assign feedback indicators to the bridge and the tracks of each turntable or transfer table. This is still supported for existing turntables and transfer tables created in **TrainController™ 5**. It is no longer possible, however, to assign feedback indicators to turntables or transfer tables newly created in **TrainController™ 7 Silver**. Existing assignments, however, can still be edited via the Turntable dialog for the time being.

Switchboards and Block Diagrams

One of the most important differences between **TrainController™ 7 Gold** and other versions of **TrainController™** is the capability to work with more than one calculated block diagram. Due to the varied advantages of calculated block diagrams and because custom block diagrams are only needed in very rare situations it is strongly recom-

mended to turn each existing not calculated main block diagram of other versions into one or more calculated block diagrams.

This is done in the following way:

- If you turned off the automatic calculation of the block diagram in the other version of **TrainController™** and created additional switchboards which associations to blocks in the main block diagram, then create a calculated block diagram for each switchboard.
- All blocks, that are associated with such switchboard will be automatically moved to the newly created block diagram. Routes, that connected these blocks in the former main block diagram will be automatically moved, too.
- Routes, that connect such moved blocks with other blocks, which remain within their current diagram, will be automatically deleted, because routes across diagram boundaries are not possible. The lost linkage across the newly established diagram boundaries must be manually restored by using connector elements (see page 50). Note also, that all schedules, that contained such deleted routes must be manually restored or newly created, too.

Train Objects and Multiple Units

In **TrainController™ 5** and **TrainController™ Silver** multiple units can be arranged only, when edit mode is turned on. For this purpose so called *train* objects are created. **TrainController™ 7 Gold** provides much more flexible and powerful features to arrange and release multiple units or other train formations at any time during operation. Train objects created in other versions of **TrainController™** remain unaffected, when a data file created with such is being loaded. Thus existing trains operate as before. It is not possible, however, to create new train objects in **TrainController™ 7 Gold**. This feature is not needed here anymore, because the possibility to arrange train sets is much more powerful and flexible. You should consider to delete your existing train objects yourself as soon as possible.

Index