

TrainPlayer 8.1 - An Introduction to Working with Linked Layouts

The purpose of this document is to summarize the updated procedure for linking two or more layouts together and to explain the use of several new scripting tools which assist with controlling multiple trains across the Linked Layouts.

It is intended that Part 1 should be of interest to a user who wants to select and drive his trains from one layout to another using the Control Panel. Whereas Part 2 is targeted towards the reader who is already familiar with controlling his trains using the TrainPlayer Scripting Language.

The new scripting tools will not function with earlier versions of TrainPlayer and you will need to upgrade to TrainPlayer 8.1 to use them. For ease of access the new script commands are also documented under the Reference Tab of the Script Central dialog in TrainPlayer version 8.1.

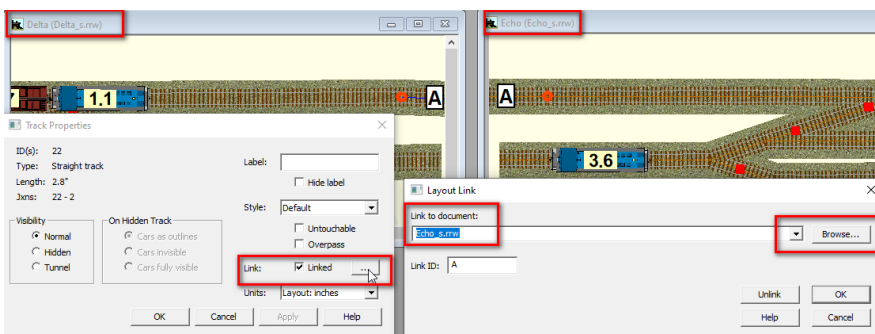
Part 1 – Linking your Layouts

Use the “Layout Link” dialog to establish a link between two layouts which have already been saved to your Layouts folder. You can’t link to a layout that only exists in memory, it must first be saved to enable the “Layout Link” dialog to find it. Several layouts can be chained together with multiple sets of links and it is also permissible to link to another spur track on the current layout.

First identify a suitable track for linking, this will need to be a spur track which is not attached to any other track at one of its two ends.

There are two ways to access the “Layout Link” dialog.

- (1) Right click on the selected spur track, select Properties, check the “Linked” box and click on the [...] browse button – see image below.
- (2) Ensure you are in TrackLayer mode, right click on the selected spur track, select “Advanced” and check the box marked “Link to layout”.



In this image we have selected the short spur track at the right end of the left image (Delta_s.rrw).

We have opened the “Layout Link” dialog from the Track Properties window and selected layout Echo_s.rrw from the drop down combo menu.

Echo was already available in the menu because it is stored in the same folder as the Delta layout we are working on (had Echo been in any other folder we would have needed the Browse... button to find it).

We then need to add a LinkID in the box below the file name. It is important to make this ID unique from any other Link IDs used on the current set of layouts as it will need to be paired with an identical LinkID on the layout we have selected to link this track to. Once this is done you can click the OK button to close the “Layout Link” dialog.

Don’t forget to click either the Apply or the OK button in the Track Props dialog to Apply this change to the layout. If your View menu “Features” selections are set to display the Track Labels you should see your Link ID on the track plan (see image). We recommend that you also click on the Save Button to lock this new data into the saved layout. So far we have only done half the job and now need to repeat this same procedure on the other layout which we are linking to.

Part 1 continued:

Once the new link is established any train passing over this spur track will automatically transfer itself to the corresponding track on the target layout. If the train is being controlled by the operator it will already appear in the Train Window and whenever it crosses to another layout the display will activate the new layout to be in front of any other open layouts. If the receiving layout is not already open then it will open automatically as the train crosses. Trains crossing links can now couple and uncouple while still bridging a link and there should be no more instances of lost or duplicated cars. You can now also drag a train across a link by pushing at one end of the train and pulling at the other.

Warning: If a train which is no longer selected in the Train Window is left running and crosses a link then it will continue to run on the target layout. However, the layout of focus will remain with the layout which contains the train currently being driven by the operator (the train displayed in the Train Window). It is the responsibility of the Operator to ensure this does not result in accidents. The safest way for a novice to operate across linked layouts is to stop one train before selecting another.

When a train arrives on a new layout the TrainID is checked. If this ID already exists on the target layout the arriving train is automatically allotted a new unique TrainID number. If any of the unique CarIDs in the arriving train already exist on the new layout the affected cars are also renumbered.

If a CarID is changed the Car Label is also updated by default. This action applies only to cars which use the default TrainPlayer AAR and CarID form of labeling. If the Car Label has been customized by the user it will not be updated but the new unique CarID will still prevent confusion with other cars on the layout.

In the image (below right) the Blue engine was originally TrainID = 1 and CarID = 1 when it was running on the left layout. Here it is halfway across a jump between two layouts.. The target layout already has TrainIDs of 1, 2 and 3 so the arriving train is allocated a new TrainID of 4. As the target layout also has a CarID = 1 on its Train2, the engine is also allocated a new CarID number of 2 (the lowest unused CarID on this layout).

The keen observer will notice an unusual format for the car label. This was introduced in TrainPlayer 8.1 to assist with development. It displays as TrainID|dot|CarID and can be useful for monitoring the progress of jumps.

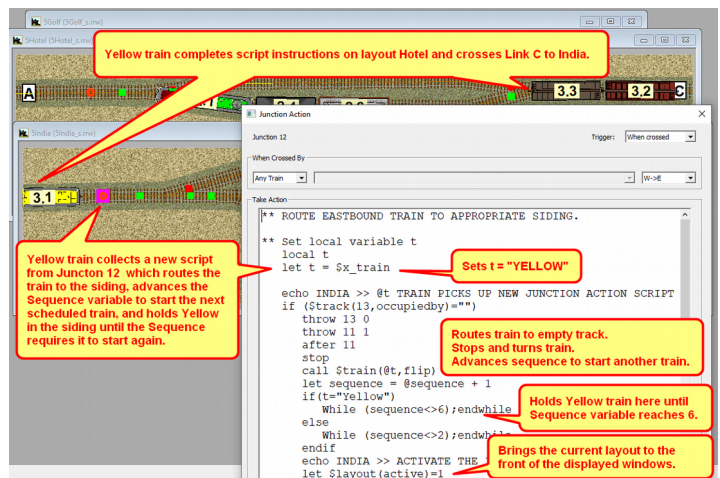
This new label format doesn't replace the original car label in memory and it only affects the display. The actual car label can still be seen by hovering over the car and viewing the tooltip. You can set up your system to display this new form of label by checking the box on the Trains Preferences Tab.

Part 2 – Script Control for Linked Layouts

TrainPlayer 8.1 makes it possible to have several scripts running simultaneously and interacting across multiple layouts.

To facilitate this all individual scripts are confined to running on a single layout and any unintended script remnants will be discarded when a train leaves the layout for which the script was written. This avoids any unforeseen consequences which might occur if an inappropriate script began to reference the wrong switch numbers on a new layout.

This means that any train crossing a link to a new layout needs to immediately collect a new set of script instructions to continue to control the train. We can achieve this with Junction Actions.



Part 2 continued:

Using the different Script Types with Linked Layouts

Master Script

A Master Script runs only once when a layout is first opened. It doesn't run again when the layout is Activated (brought to the front of the displayed layouts). So if you want to reactivate an open layout and rerun its Master Script you should use the "file open" command or "Revert to Saved" command.

Master Scripts can be used for setting up Global Variables you need to choreograph your scripts, for starting the trains moving and for applying initial Drive blocks to your trains. They can also be used for opening other layouts required for the scripted sequence, or for activating which layout should be displayed in front of all the other open layouts.

Train Script

Train Scripts are best confined to the local trains which will not cross any of the links to other layouts. Remember that to avoid cross contamination these Train Scripts will be discarded if the train they are attached to crosses a link to another layout. This means that they will not be available to rewind even if the train eventually returns to its home layout.

Junction Action Script

Junction Actions are the recommended method for controlling trains across multiple layouts. The ideal location for placing these scripts is on the junction that connects the designated linking tracks to the other tracks on the current layout. You should ensure that the trigger direction is correctly set to apply the script to trains moving **onto** the target layout.

Unless the train is intended to terminate on the target layout, each script should end with a command which routes and drives the same train across a link to another layout. These scripts can include Drive blocks to transfer control to other trains on the same layout. They can also be used to modify the values in Global variables which may be being monitored by the scripts which are running on other layouts in the linked set.

Panic Button

The "**Stop all scripts**" toolbar button now works over all open layouts.

This instantly stops all the scripts that are running across all the open layouts.

It also instantly stops any trains that are still moving even if their scripts have already expired.

Extended Script Menu Commands

File open <filename> now accepts a filename or filepath argument. If this is supplied the file dialog doesn't come up and the specified path is used. Where a simple filename is used the program will assume that the required layout will be found in the same folder as the calling rrw. If the file is to be found in a different folder you can use a relative path to the rrw which is running the script.

Example from Master Script of 5Golf_s.rrw:

```
file open 5Hotel_s.rrw  
file open 5India_s.rrw
```

File "Save as" <filename> When an argument is supplied the file is saved to the path provided without opening the "Save as" dialog. If a filename is supplied without a path the file is saved to the same folder as the layout running the script.

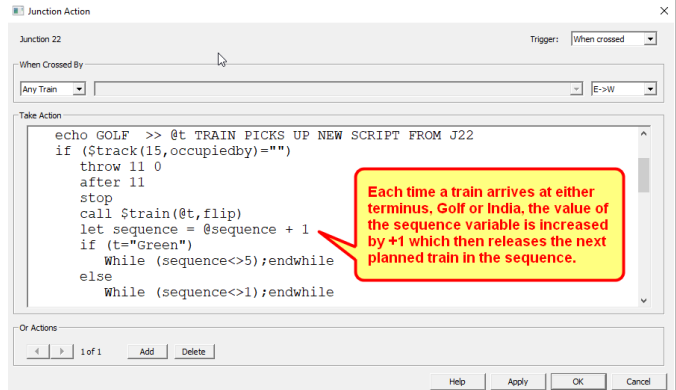
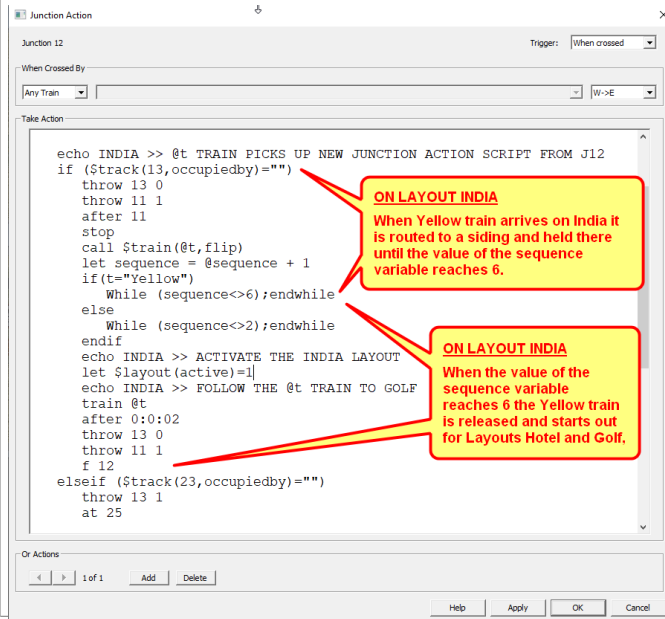
File "Save all" is used to save multiple open documents to their original folders

Making use of Script Variables

Script Variables are global and can be used to pass information between the scripts on different layouts. This makes it possible to hold up one train on a specific layout and only release it when another train has performed a specific action on a different layout. This is demonstrated on the 5GHI linked set.

The Yellow train arrives on Layout India as Train1 and waits there so that it can return to Golf later when the Sequence number reaches 6.

On 5GHI the trains run as pairs in opposing directions, crossing each other on Layout Hotel.



When the Steam Train reaches India and the Passenger Train reaches Golf (in either order) the sequence count advances, first to 5 and then to 6.

This simultaneously releases the Green Train from Golf and the Yellow Train from India both of which have been waiting for this condition to be met.

New Registry Keys

<p>LinkLabelFont</p>	<p>Stores details of the font used on the extra Track Label which identifies the stub tracks on Linked Layouts. This is available for both Read and Write.</p> <p>Uses the format 110,700,0,0,Arial this is the default setting. Format of string is: "height,weight,italic,color,faceName" Weight 0 is normal, 700 is bold, Italic is 1,Color 255 = Red.</p>
<p>CarIDsOnTops</p>	<p>0 = Car top displays the defined Car Label This is the Default Setting. (Displays either the TrainPlayer AAR/CarID combo or a customized label). 1 = Car top displays TrainID.CarID. (The new format for monitoring the automatic renumbering process)</p>
<p>LabelUpdateOnJump</p>	<p>The feature to automatically update the default AAR CarID style of Car Label, to avoid car label duplications, can be disabled in the Windows Registry.</p> <p>This dictates whether labels are automatically updated when car tags change on jumping. This is available for both Read and Write. Default is 1 = True.</p> <p>To change this for a single layout use: <code>Set LabelUpdateOnJump 0</code> To change permanently and retain your original CarIDs edit this in the registry.</p>

Part 2 continued:

New Arguments for existing Script Functions

\$layout(layoutID)

Linked layouts have this additional property known as the **LayoutID** which is automatically applied by TrainPlayer. It can be modified in the Layout Properties dialog but you wouldn't normally need to change this. The default LayoutID normally comprises the first two letters of the layout name followed by a unique integer added by the program.

On opening any linked layout, its LayoutID is checked against other open layouts; if there is a duplication, an alert comes up and offers to change the LayoutID of the layout being opened.

The \$layout function now gives script access to the LayoutID for both read and write.

Example: Set \$layout(layoutID) IN25

\$layout(active) This function gives script access for both read and write.

Returns 1 if the layout running the script is the Active foreground layout, otherwise returns 0

Example: Set the layout running this script as the Active foreground layout.

Set \$layout(active) 1

alternatively

Let \$layout(active) = 1

Note: You can't set this value to 0, instead you must bring one of the other layouts to the foreground. To facilitate this the function will accept another layout name as a second argument. If the filename contains spaces it should be quoted.

Example: To bring an open layout from the background to the foreground

let \$layout(active,India.rrw)=1

Note: It is usually necessary to select a new train for the train window from those available on the newly activated layout after using this command – see **Train Trainname** below.

\$car(id,layout) Returns the name of the layout containing the specified car.

\$car(id,layoutID) Returns the LayoutID number of the layout containing the specified car.

\$train(id,layout) Returns the name of the layout containing the specified train.

\$train(id,layoutID) Returns the LayoutID of the layout containing the specified train.

Train TrainID Selects the specified train by its ID number and places it in the Train Window

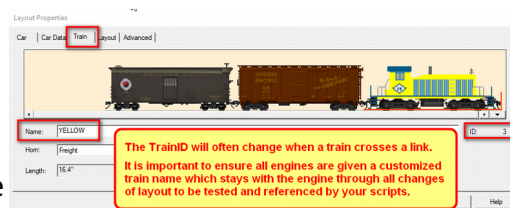
Once selected in the Train Window the train will be followed from layout to layout until another selection is made. Use this with caution as TrainID numbers are automatically changed by the program to avoid duplicated train references.

We suggest giving all trains with engines a customized Name and using the **Train Trainname** command to select your trains (see below).

Train Trainname Selects the named train and places it in the Train Window

Once selected the train will be followed from layout to layout until another selection is made.

Important: We recommend giving all the trains across your Linked Layout domain customized names which you can refer to when writing your scripts. These names will stick with the relevant engines no matter how often the TrainID or CarID is changed by TrainPlayer to avoid conflicts with other trains in use on the different layouts.



Part 2 continued:

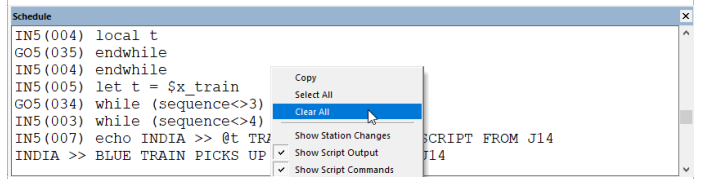
Schedule Window Upgrades

Clear All

The Schedule Window no longer clears itself automatically when a new layout is opened.

You can now keep track of the output from all layouts which are simultaneously running.

To clear the Schedule Window at any time you should right click on the text and select "Clear All". Alternatively the window can be cleared by inserting a CLS command in your script.

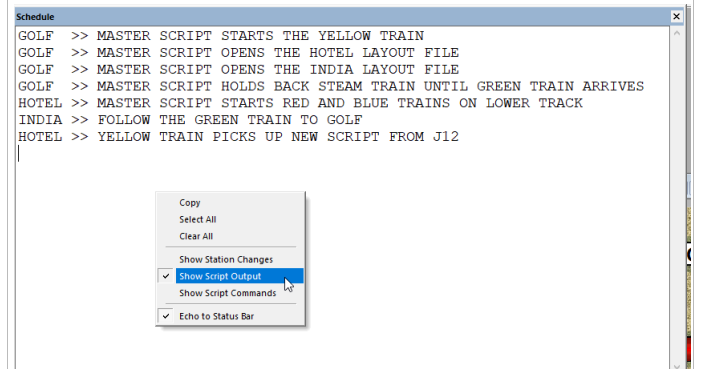


Show Script Output

A recent bug which prevented Comments being echoed to the Schedule Window has been fixed.

If the "Show Script Output" option is checked then all Comments (except for those starting with **) will be echoed to the Schedule Window along with the Echo statements.

The example image on the right does not contain any Comments and is made up solely of Echo Statements.



Show Script Commands

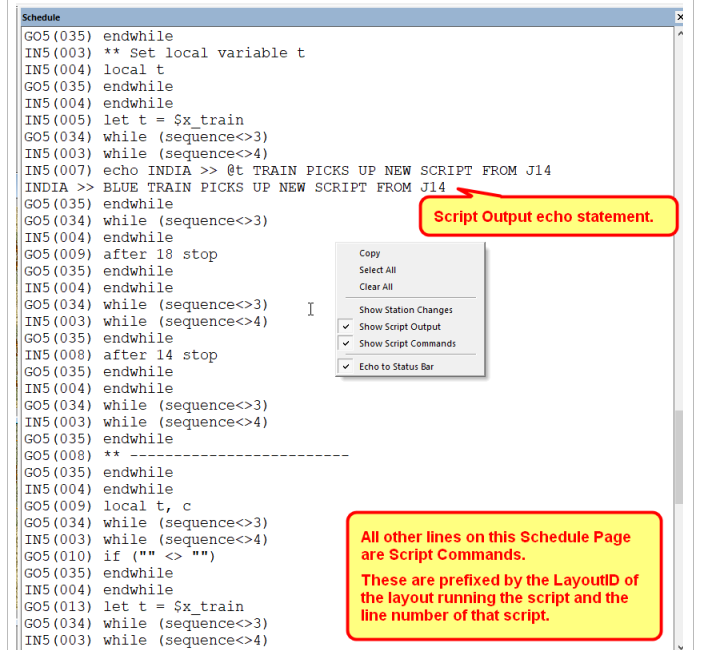
A new Option on the Schedule Window context menu permits the Script Commands to be echoed to this window as they are executed.

This is a new and preferred alternative to following the progress of a script in the Script Editor window where Subroutine and Junction Action code is not displayed in the window.

With the Schedule Window open and the Show Script Commands option checked, every command is echoed to the Window as it is being executed.

Each line of script code is preceded by the LayoutID number and the line number of the script being executed.

The resulting Schedule Window output can then be copied to a text file for further analysis if necessary.



Document compiled from information supplied by Jim Dill during the development phase of Trainplayer v8.1

The 5Golf, 5Hotel and 5India layouts can all be found in the "Scripts and Puzzles\Scripting examples" folder of your Layout Chooser. To run these layouts just start up 5Golf and the other layouts will open automatically.

Richard Fletcher, July 2021