Documentation for Drive to trackmarklist command

The « Autodriveto Ttrackmark list » is a driver command that enables :
- to autodrive to all trackmarks in a list of trackmarks
- to autodrive to the first available (not occupied by a train) trackmark in a list of trackmarks
- to autodrive randomly to any of the available trackmarks in a list of trackmarks

The driver command works by replacing itself by a Autodrive command (from Brummfondel) to drive to all or to the selected available trackmark.

1. Required components

Depending on the type of driver command to generate, the following drivers commands may be required :

Autodrive (from Brummfondel)	< kuid:192081:4:3>

This command is available on DLS. You need to download it before using the Autodrive to Trackmark list driver command.

2. Trackmark list definition

In order to use the driver command, you need first to define some sets of trackmarks as trackmarklist.

To define a trackmarklist, you can use 2 methods :

- use the implicit by defaut naming convention

  in this defaut naming convention, all trackmarks in a set should follow the naming convention :
  trackmark list group name(xxxx) or
  trackmark list group name[xxxx]

  where trackmark list group name can be any name
  and xxxx can be any string.

  The system recognizes the trackmark as included in a trackmark list group name by

the fact it has an index between parenthesis or beackets.

for example, here area trackmark list examples :

Port Ogden(01)
Port Ogden(02)
Port Ogden(AA)   belongs all three to the Port Ogden trackmark list group.

Johnsburg Station[P1]
Johnsburg Station[P2]
Johnsburg Station[P3] belongs all three to the Johnsburg Station trackmark list group.

You should also notice that when using implicit defaut naming convention, the trackmarks names in a group are sorted by alphabetical ascending order.

- use arbitrary names with no naming convention (can be useful to reuse trackmarks allready defined ...).

this needs a little more definition through session parameters variable (to be defined by using session parameters rule from pguy - <kuid2:61392:1024:1>)

for each trackmark list group, you need to define a session parameter variable following the naming convention :

tklgroup:xxxxxx

with a string value of

groupname(trackmark1;trackmark2;trackmark3;...;trackmarkXX) or
groupname[trackmark1;trackmark2;trackmark3;...;trackmarkXX]

the text xxxxxx after tklgroup: can be any text as long as the name tklgroup:xxxxxx is a unique session parameter name. the string value xxxxx is not used by the command.

Groupname will be the name that appears under the driver command menu and can be freely choosen.
Trackmark1;trackmark2;...;trackmarkNN is the list of the trackmarks name, separated by a semicolon.
The trackmark names are separated form the group name either by parenthesis or by brackets.

Example :

session parameter variable   tklgroup:group01 with value
Port Ogden Platforms(Port Ogden – P1;Port Ogden – P2; Port Ogden – P3)

will define a trackmarklist group whose name is « Port Ogden Platforms » made of three trackmarks « Port Ogden – P1 » , « Port Ogden – P2 » , « Port – Ogden – P3 ».

You should notice that when using explicit naming by defining session parameters, the order of the trackmarks is kept. No sorting is done on the trackmarks names.

As the command is replaced by an autodrive command from Brummfondel, you need to set the pathcontrol before autodriving. If you use the PathTrigger rule from Brummfondel to setup the path, you can only use trackmarklist definitions with paranthesis as brackets have special meaning in the pathtrigger rule. Use trackmark selection to select the correct trackmark with paranthesis under PathTrigger rule parameter setup.

3. Menu interface

The driver command « AutodriveTo Trackmark List » offers 3 options at first menu level :

AutodriveThrough all trackmarks in trackmark list

AutodriveTo 1st available trackmark in trackmark list

Autodrive to random available trackmark in trackmark list

The second menu level will list all avialable trackmark list group names in order to choose the target group for the command.

4. How a trackmark is checked as available and some limits of this driver command

To check the avaibility of a trackmark, the command searchs the unique trainzobject id ot the track object associated with the trackmark.

Then it loops first on all trains in the route and on all vehicles in each train and search the id of the track associated with each vehicle.

If some vehicle are on the same track as the trackmark, the trackmark is considered as occupied. If no vehicle on any train is on the same track as the trackmark, the trackmark is considered as available.

The method is very very basic and not optimized ... so the command can be sensitive to the number of trains and vehicles in the route.

Another limit is that the command has no memory. If two commands ask simultanously for the 1st avaialable trackmark on the same trackmarklist, they will obtain and drive to the same trackmark destination. The command looks only to the current avaibility of a trackmark and do not keep in mind that it has allready given this available trackmark destination to another driver ... Still some work for algorithms improvment to introduce trackmark reservation ...

The best method to reduce the risk of two trains going to the same available platform is to put a trackmark just in front of the junctions diverging to the platforms and to put trackmarks for each platform in the same trackmark list group.

Then you can schedule a train with two order : first one a DriveVia command to the trackmark before the diverging junctions and then a second command DriveTo 1st available trackmark in the group. So the choice of the available trackmark will be delayed until arriving to the diverging junctions.

Hope this helps for scheduling train to available trackmarks.