Project:	**_Traffic Simulation Software_**		Document No.	Issue:
Title:	OpenDRIVE Scenario Editor - User Manual		VI2008.027	M
	Date:	15.01.2016	no. of pages:	39
	Issuing Party:	VIRES Simulationstechnologie GmbH		
	Author:	Andreas Biehn		
	Distribution List:	public		
		VIRES	M. Dupuis W. Karl D. Wiesenhütter A. Biehn	

Version Control

date	issue	name	remarks
11.03.2008	A	Dupuis	derived from "old" version
07.05.2008	B	Knerr	Version 0.7.2
29.08.2008	C	Knerr	Version 0.7.5
08.02.2010	D	Knerr	Version 0.9.0
15.07.2010	E	Knerr	Version 0.9.1
31.05.2011	F	Knerr	Version 0.10.1
17.07.2013	G	Wiesenhütter	Version 1.4.1
30.09.2013	H	Wiesenhütter	Version 1.4.2
18.02.2014	I	Wiesenhütter	Version 1.5.0
18.06.2014	J	Biehn	Version 1.6.0
30.09.2014	K	Biehn	Version 1.7.0
22.05.2015	L	Biehn	Version 1.9.0
15.01.2016	M	Biehn	Version 2.0.1

Table of Contents

1   Scope

This document provides a user-level description of the functionality implemented in the VIRES traffic simulation system and its graphical user interface *Scenario Editor*.

2   Scenario Editor

The "Scenario Editor" is the tool to create and modify repeatable executable setups of scene, traffic and various kinds of deterministic actions. We call such a setup a **scenario**. Deterministic actions vary from traffic centered ones to external notification via a user extendable network interface. All scenarios are written in human readable XML format and may also be edited via any text editor, although this is only recommended for "power users".

In the following sections all necessary steps to implement a scenario and the possibilities of configuring the editor by the user are presented.

2.1   Options of the main window

2.1.1   Main menus

File Edit View Extra

- The menu item *File* provides action buttons to create new, open, save, reset scenarios, to recover accidentally overwritten scenarios and to close the editor. Resetting means to get the state of the scenario that was last saved to disk. Recovering in contrast rereads the file saved to disk before the current (if there is any). If enabled these files are created automatically by the editor showing a tilde symbol after the filename. All actions that could possibly compromise a currently modified scenario (like reset, open, new, recover) lead to presenting a dialog to the user asking how he would like to proceed.

- Menu item *Edit* includes undo/redo functions, which take effect on all user input regarding the scenario itself like player positions and attributes, even done through sub-dialogs. The undo/redo stack gets reset for every new loaded scenario or after reset and network mode. This feature is currently disabled.
  In this menu the scenario properties, traffic lights and editor settings dialogs can be opened.

- Under menu item *View,* additional graphical data can be enabled or disabled for the road window:

  - road signs
  - traffic lights
  - sign references
  - player names
  - vehicle lights
  - pulk (swarm) traffic region around central player
  - database grid (x and y offset can be changed separately via editor settings)
  - database origin
  - display of lanes marked as border (used for some simulators)
  - drawing of database
  - drawing of road objects
  - user defined annotations
  - draw path

Date:	15.01.2016	Title:	User Manual			
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 4 of 39	

- draw path shapes
- draw rail paths
- highlight railroad switches
- hiding of selected elements (will then be grayed out and not selectable)
- showing of all hidden elements again

Menu item *Center View* resets the view of the road network to be centered and scaled in a way that the whole database is visible again. The scenario monitor and the action info monitor can also be started by this menu.

Using the last four checkboxes the status toolbar, the main toolbar, messages and SCP GUI dock windows can be shown or hidden.

• Menu *Extra* holds a dialog to translate all elements of the current scenario like players, characters, actions and absolute waypoints in space. This is useful, if the origin of the road network has changed and the scenario content shall be adapted easily.
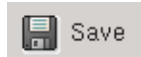
2.1.2 Icons

New creates an unnamed scenario with no content.

Open loads predefined scenarios via a file chooser dialog.

Save allows the user to define a filename for a scenario and save it to disk.

Reset configures the scenario as it was defined in the last saved state.

Properties opens the scenario properties dialog (please read next sub section).

*Traffic Lights* opens the dialog for configuring the traffic lights and phases.

Network mode provides a bird view functionality that allows the tracking of all moving players and dynamic signals of the running simulation including the external simulator vehicle (mock-up).
This button opens a menu where the user can choose to disable or enable the network mode permanently as well as letting the editor automatically connect to or disconnect from running simulations.
Option *Auto reset* reloads the scenario as saved after a simulation was stopped.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 5 of 39

Copyright note according to DIN 34.
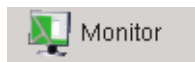
	Monitor opens a special dialog providing the user with detailed data about a selected player.
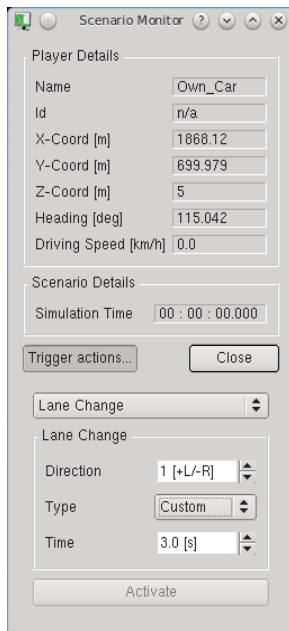
The user can pick a player with the usual edit tool and is then given information about the selected player's name, id, current coordinates and orientation.

Below the player details there is a digital clock showing the current simulation time provided that a simulation is currently running and the editor is connected to the network.

Pressing the button *Trigger actions,* the monitor dialog expands to a real-time remote control of the player. In the case of a running simulation and being connected to the network all available actions can be triggered for the selected player. Please refer to the chapter "Action-Control Concept" for details about the actions, their parameters and effects.

	Action Info Monitor opens the dialog for the monitoring of started and completed actions. Please consult chapter 2.10 for an explanation.

2.1.3 Navigation Bar
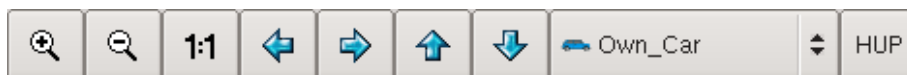
By means of the navigation bar the user can zoom the road map, use the detail zoom factor (1:1) and move the map view in west / east / north / south direction.

In the drop-down menu all players of the scenario are displayed by name. External players are sorted first, indicated by a blue colored icon, internal players are sorted subsequently, indicated by a red colored icon. Selecting one name will center the view on this player. During bird view mode a chosen player stays in focus even when being in motion.

The last button of the navigation bar is enabled when a player is selected as central player. If toggled, the view on the road network is always rotated with the motion of the central player in such a way that his heading orientation will be the up orientation of the view.

2.1.4 Manipulation Tools

	Select one element by single mouse click or many elements by dragging a rectangle around the elements (players, characters, objects, actions, waypoints). Single elements can be added to the selected elements by holding *Ctrl-Key.* In this way already selected elements can also

be removed from the selection by a single mouse click on the element. The selection of many elements is different for the case the properties window is open. In this case, only single elements can be selected with a click or added to the selection with the help of *Ctrl-Key*.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 6 of 39

Activating this tool, a new player is added to the position on the map by clicking subsequently. If the position lies on a road, the editor automatically determines the orientation of the player from lane information and traffic direction (right-hand / left-hand traffic setting). A new player is automatically named "New Player" plus a running number and selects the first available vehicle and driver definitions by default.

Activating this tool, a new character is added to the position on the map by clicking subsequently. The Z-coordinate (height) will be determined by the editor from the road height if available and is updated every time the character gets moved. The default orientation for characters is always zero degree / positive X-axis. A new character is named "New Character" plus a running number by default and selects the first available character type and model.

Activating this tool, a new object is added to the position on the map by clicking subsequently. The Z-coordinate (height) will be determined by the editor from the road height if available and is updated every time the object gets moved. The default orientation for object is always zero degree / positive X-axis. A new object is named "New Object" plus a running number by default and selects the first available object definition.

This tool is enabled, when a player, a character or a object is selected and adds a new trigger for actions to them at the position on the map where the user clicks to. By default new triggers are absolute positioned and do not contain any enabled actions.

Pressing this tool button the waypoint path dialog for routes / paths on the road network is opened. To create a route the user just has to click on a position on the road network where he wants a route to start. This first waypoint is then displayed as number one in the center of the track. The whole track itself is drawn yellow marked to indicate which part of the road network has already been selected. The current status and length of the path is indicated by the dialog in the path details section. A path with less than two waypoints will be marked as incomplete in yellow color and not be accepted by the dialog. As soon as a path is complete and does not contain any errors, status changes to *OK* with a green background. Each path must be given a unique name to be able to easily recognize the path later-on which is controlled by the editor. Duplicate names are not accepted and highlighted with a red background.

Because moving and zooming tools of the main window stay enabled, the user can go to any other position on the road network and select them as waypoints for the path. In this mode the mouse cursor changes to to clarify add-mode. The user can always watch online the path being created whereas the selected tracks are marked solid yellow and automatically found tracks are marked transparent yellow.

If a path between waypoints can not be resolved or if there is an error in the logical database, the first waypoint causing the error will be displayed in the red-colored status line. Junctions are excluded from being selected as waypoint, because of the possibly resulting ambiguity of the further way.
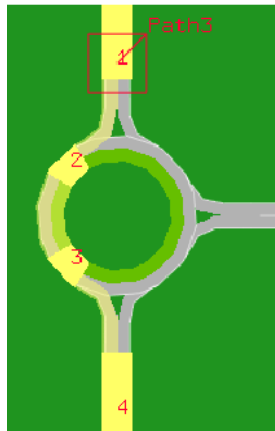
Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 7 of 39

Copyright note according to DIN 34.

In the waypoint track list all waypoints appear as selected by the user. Right clicking on a waypoint entry in the waypoint list centers the view on that waypoint. Single waypoints can be selected and deleted, which results in a new calculation of a path from the remaining waypoints. Selected waypoints are drawn highlighted on the road. With the up and down buttons the order of the waypoints in the path can be changed; new waypoints are always inserted after the selected one.

If the user started creating a completely new path, the title of the dialog shows "create waypoint path". In this case the path has to be added to the list of all paths by pressing the *add new path* button, otherwise the creation can be aborted by pressing escape or the *cancel* button. Paths containing errors are refused from being added to the list.

The dialog then changes from create- to edit-mode, which means that the waypoints can still be changed like before, but additionally be moved by mouse. If the user does not select an existing waypoint, but clicks a valid position on the road, a new waypoint is added to the waypoint list. Using button *New Path* the process of creating a path can be started again.
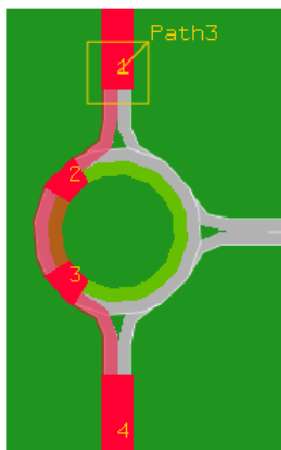
The current path, i.e. the path whose name is displayed after *identifier*, can be removed from the scenario by pressing the *Delete Path* button. It is only enabled in cases where a delete is possible.

Button *Jump To Paths* centers the map view in a way that all marked paths from the list are shown. The view is only updated on button toggle or path selection to not disrupt the user working on paths.

All marked paths are painted in transparent red color, which also applies for the current one after closing the dialog.

Double clicking any waypoint in the map opens the dialog again with the related path set as current.

Furthermore the assignment of options for waypoints is possible. The user can choose between:

- **classic:** a path that is only completed up to the next waypoint if there is not more than one junction in-between. This option can not be combined with any other.
- **shortest path:** predecessor waypoint is connected to this waypoint via the shortest possible path
- **fastest path:** connection via fastest possible path regarding length and road type entries, but not regarding waiting times at junctions or traffic signs
- **fewest junctions:** connection via fewest junctions, every junction counts the very same

If two possible paths have exactly the same score it depends on the logical database which alternative is chosen. In fact, at intersections most of the time the first connection stated in the database wins.

Paths beginning and ending at the same coordinates (less than 0.1 m deviation) are handled as "closed paths" if End Action "loop" is used. That means looping is not performed as position jump but as smooth transition, keeping current lane offset etc.
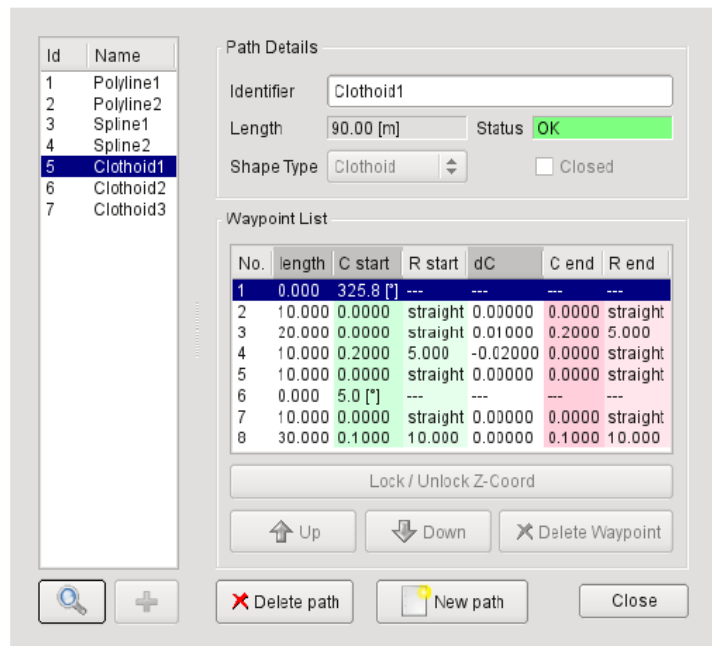
Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.:	Issue:		Page:
		VI2008.027	M		8 of 39

This tool button opens the path shape dialog. This dialog works much like the waypoint path dialog: it presents an editable input box for a path shape name and a line showing the current status as well as the length. Besides, each shape has one of the three potential types (polyline, spline, clothoid). Checkbox *Closed* determines if a shape is automatically constructed closed; this option is not available for clothoid shapes. A path shape needs at least two waypoints to be accepted by the dialog. Maneuvering the main view the user can click on road positions that are then selected as waypoints. Even positions outside the road are accepted. In case they are located too far from the road to determine the world's height coordinate an info message is shown.

A waypoint is a coordinate in 3D-space clamped to ground by default with an orientation. The dialog tries to determine height from road data. To prevent the height value from being changed when moving a waypoint, the z-coordinate can be locked. Locking is indicated by bold type for the z-value and the display of a small lock next to the waypoint icon. Nonetheless if e.g. pavement height is not specified in the road description or other coordinates shall be edited without moving the waypoint by mouse over the map, the user can adjust all space coordinates by double- clicking the respective table cell.

The shape will always run through each waypoint in the direction shown by its orientation (the small red arrow at one edge of the icon).

All waypoints are listed, whereas single ones can be selected, moved up and down in the list (only polylines) or deleted which results in the calculation of a new path shape. There is a preview of the resulting shape in the main window using a dashed yellow line and drawing marked waypoints highlighted by a red box around. New waypoints are always inserted after the current selected one. In case the dialog is ready to create new waypoints the mouse cursor looks like this: If the cursor looks like a snapping hand the waypoint residing underneath will be selected. In case of a clothoid shape the waypoints in the list stand for segments of that shape, which will be described in detail below. Adding a waypoint then means adding a default straight segment of 10 meter length.

Closing the dialog marked shapes are painted in orange dashed lines. When the dialog is shown, new shapes or current shapes in modify mode are painted yellow.

Date:	15.01.2016	Title:	User Manual			
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M			Page: 9 of 39

Copyright note according to DIN 34.

All shapes can be used as intended trajectory for a driver; nonetheless there are some important differences between the shape types:
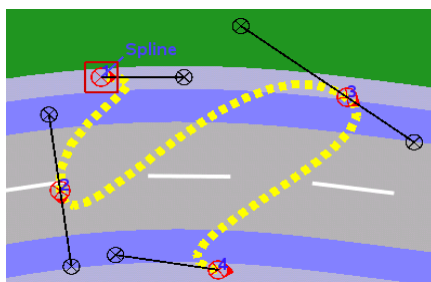
- **Polyline:** This is the default shape. All waypoints are connected with straight lines, therefore curvature over the whole shape is constantly zero. Every waypoint's height can be adjusted and
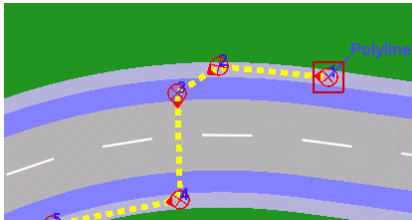
No.	X	Y	Z	Yaw
1	-589.874	2716.359	9.620	174.29
2	-600.746	2717.447	9.620	212.74
3	-604.897	2714.778	9.500	271.62
4	-604.600	2704.301	9.620	191.05
5	-617.251	2701.831	9.620	191.05

defaults to road level. Moving a waypoint may change the orientation of its neighbor waypoints. For a closed shape, first and last waypoint are simply connected with a straight line and length increases about their distance. This shape can be used to animate characters. Depending on the underlying library character paths may be smoothed.

- **Spline:** The user sets waypoints through which the shape must run. Each waypoint (except first and last) has two control points drawn black. They are connected to each other with a straight line and display the tangent trough the waypoint. The underlying spline

No.	X	Y	Z	Yaw
1	-605.242	2717.109	9.620	0.00
	-600.242	2717.109	9.620	0.00
	-608.409	2714.668	9.620	0.00
2	-607.729	2709.715	9.500	277.82
	-607.048	2704.761	9.620	0.00
	-596.812	2720.445	9.620	0.00
3	-590.388	2715.801	9.620	324.13
	-586.337	2712.871	9.500	0.00
	-604.377	2705.513	9.500	0.00
4	-598.175	2704.546	9.620	351.14

consists of cubic polynomials and is constrained to be continuous. In the waypoint list all real waypoints are numbered and highlighted red, whereas control points are listed regularly. Control points are used to modify the shape in-between two waypoints and are moved together with their parent waypoint. Control points can not be deleted individually. The 'closed' option connects the first and last waypoint on a direct but continuous way. This shape can also be used to animate characters.
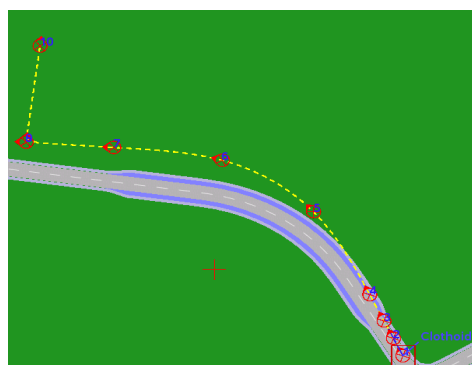
- **Clothoid:** This shape works differently than the other ones. The user defines segments that can be straights, clothoids or curves by specifying the length, the curvature at the beginning the segment and the curvature change over the whole segment. In this case the user sets the first waypoint by clicking into the map as usual. All subsequent clicks create a

No.	length	C start	R start	dC	C end	R end
1	0.000	115.6 [°]	---	---	---	---
2	10.000	0.0000	straight	0.00000	0.0000	straight
3	15.000	0.0000	straight	0.00000	0.0000	straight
4	50.000	0.0000	straight	0.00022	0.0111	90.000
5	50.000	0.0111	90.000	0.00000	0.0111	90.000
6	50.000	0.0100	100.000	-0.00022	-0.0011	-900.001
7	40.000	0.0000	straight	0.00000	0.0000	straight
8	0.000	82.7 [°]	---	---	---	---

of

default ten meter segment. In the list all segments are shown as a table and values can be changed by double-clicking and editing a cell. To simplify the process of shape creation also the calculated end curvature is shown and can be edited. Modifications recalculate the curvature change. Additionally curvatures are also displayed as radii, which are easier to handle for humans. Modification of radius recalculates the associated curvature. In the header of the table the dark colored columns mark the basis for the calculation of the segments. In the table itself the green columns mark starting curvature / radius and the red columns mark end curvature / radius. It is possible to create discreet jumps in the shape by defining a segment with null length. For such a segment the direction to continue can be specified as angle to the X-axis in degrees. Clothoid shapes can neither be closed nor used as character path shapes.

2.2 Preparation of Scenarios

For the creation of a new scenario, first the **Scenario Properties** which are accessible through the button *Properties* have to be adjusted. There are several sub-windows whose settings take effect globally on the scenario. These will be described in the following paragraphs.

With the **Scene** window the logical description of the road network (*Layout File*) and the graphical representation (*Visual Database*) of the course can be selected. The buttons "..." in the right column open dialogs to browse the file system and choose files. Both file paths can also be edited directly.
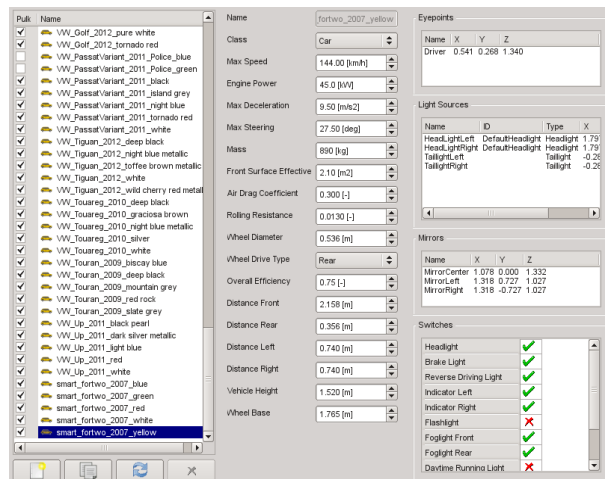
Tooltips above the respective edit lines show the path to the file as resolved by the internal file finding routine. A file that could not be found or is otherwise invalid results in an edit line highlighted in red.

The window **Vehicle Definitions** allows for the definition of different vehicles including technical specification. Those definitions have to be named uniquely to be referenced later during the creation of

players. It is recommended to use physically correct values, because the parameters directly take effect on the vehicle dynamics of players in autonomous mode. The values for the dimensions of the vehicle are used for distance calculation among the players and therefore are important for externally controlled and moved players, too. Every vehicle can be used for swarm traffic if the checkbox on the left side of the list is set, otherwise it will be excluded from being a swarm player.
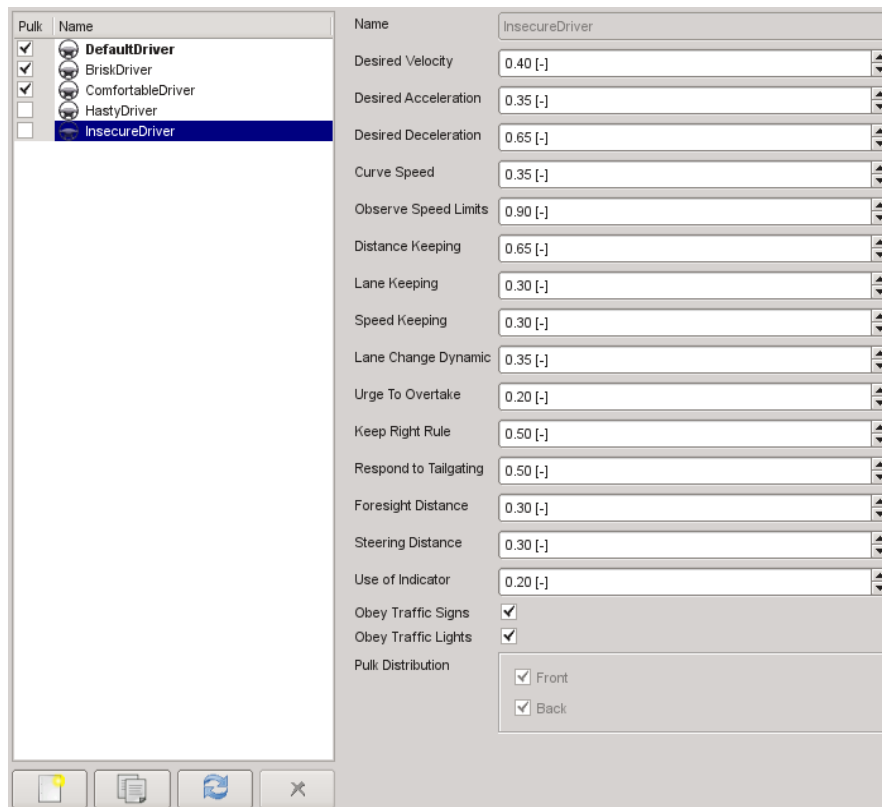
The right hand side shows information about the 3D-model. Positions of eyepoints, light sources and mirrors are displayed. The switches table shows which features of the vehicles 3D-model can be switched on/off.

If there is a vehicle database referenced by the editor, the list is already pre-filled after the start. Any change of properties to database entries is marked with bold type. If there is at least one difference to the database the name of the vehicle will be written in bold type, too.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 11 of 39

Copyright note according to DIN 34.

With the dialog **Driver Definitions** the user is able to modify and create different types of drivers by means of some important selected behavior characteristics. If no other drivers are defined, all newly created players automatically reference the default driver, which represents a standard average driving characteristic. If there is a driver database available there can be additional definitions listed. Changes to database entries are highlighted in bold type similarly to the vehicle database.

In the following there is a description of the attributes and their values.
In general all values are normalized to be between zero and one to make it easier for the user to modify selected behavior without having to know all underlying physical or psychological thresholds. 0.0 always means to have very little or no portion of that characteristic, 0.5 would be an average driver and 1.0 means to have the highest possible influence of that characteristic.

Desired Velocity: uninfluenced highest speed a driver will aim to go, if there are no other limits. Ranges from driving very slow to highest vehicle speed.

Desired Acceleration: dynamic of how to change to higher speeds. Ranges from almost no acceleration to the maximum acceleration of the vehicle type.

Desired Deceleration: dynamic of how to change to lower speeds (not for emergency braking because of leading vehicles). Ranges from almost no deceleration to uncomfortable high.

Curve Speed: specifies the maximum speed the driver tolerates for curves. Ranges from anxious driving to racing car limits.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 12 of 39

Copyright note according to DIN 34.

Observe Speed Limits: defines the amount of observing implicit or explicit speed limits. Ranges from totally violating limits to 100% respecting speed.

Distance keeping: defines the desired time gap to leading vehicles from driving extremely close to holding very much distance.

Lane keeping: defines the ability to stay at the center of the lane while driving. Ranges from oscillating between the lane borders to keeping the center exactly.

Speed keeping: defines the ability of the driver to exactly move at his desired speed. Ranges from never really matching the speed to exactly holding it.

Lane Changing Dynamic: the way an autonomous driver tries to change a lane. Ranges from slow to high lateral speeds during the lane change.

Urge To Overtake: combines several criteria for when a driver autonomously wants to perform overtaking. A driver with this value being 0.0 never tries to overtake at all. Value 1.0 results in an agitated overtaking behavior.

Keep Right Rule: determines the way a driver chooses to change to a free side lane. If 0.0 he will not autonomously change to the side, if 1.0 he will change as early as possible.

Respond To Tailgating: the way a driver is being influenced by adjacent driving vehicles. Ranges from not responding to even very close vehicles to responding early.

Foresight Distance: the preview distance the driver looks ahead for other vehicles and signs. Ranges from being very short-sighted to being anticipatory.

Steering Distance: the distance that the driver aims for steering. Ranges from steering to proximate points and therefore indicating late at intersections to steering to far distant points resulting in cutting corners and indicating early.

Use of Indicator: Defines the time the driver uses indicator for signifying turning left or right. Ranges from never using the indicator to using the indicator a long time ahead the planed turn.

Obey Traffic Signs: switches observance of right of way indicating signs on or off

Obey Traffic Lights: switches observance of traffic lights on or off

Pulk Distribution: If the Pulk checkbox in the driver list and the Front and Back checkboxes in the Pulk Distribution group are checked, that driver is distributed randomly to pulk vehicles. By deselecting one of Front or Back, one can enforce distribution of the selected driver only to vehicles which are set up in front of or behind the central player.

The entries in **Traffic Elements** allow for the definition of various times for lane changes (i.e. lane changes with different duration) and for symbolic names associated with the different definitions. The symbolic names may be used directly for lane change actions.
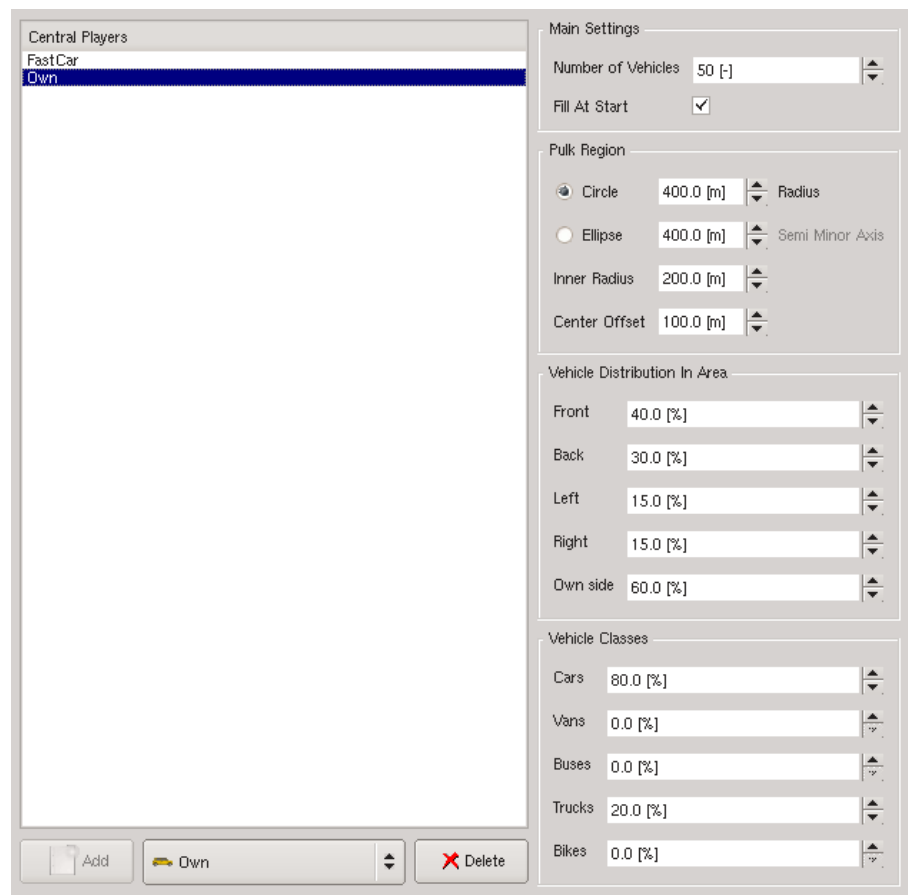
The window **Counters** provides the possibility to define various variables for the simulation used e.g. as lap counters. All counters are generally instantiated with value zero and can be incremented, decremented or set during runtime by special actions. Counters get initialized at each start of a simulation.

The tab **Pulk Traffic** allows the user to define so-called swarm traffic around one or more central players. This special kind of autonomous traffic is automatically removed from the roads if it is leaving the pulk region around the central player and then replaced to another position at the edge of the region. Every player predefined in the scenario can be the central player of a swarm. A region around this player can be specified (*Pulk Region*) as a circle where the radius of the region is to be given or as an ellipse where semi-major and semi-minor axis can be defined. The number of vehicles driving in the swarm is only limited by the maximum number of players allowed for the simulation in general.

Using *Vehicle Distribution in Area* hints can be given to the algorithm where to preferentially place vehicles; the box *Own side of road* adjusts placing of vehicles between traffic using the same side of road and oncoming traffic. Pulk vehicles are only added to the area between the radius and the inner radius. If this area is very small, due to the little space, less vehicles than defined will be positioned.

It is possible to define the distribution of different vehicle classes like 'car', 'truck' or 'motorbike' for the swarm. The vehicle definitions of the players automatically set-up by the traffic manager in the pulk are taken out of the pool of user-defined vehicle definitions as described above. If a vehicle of e.g. type 'truck' was not specified the manager uses one of the other defined types as a replacement (if the checkmark in vehicle definition is set).

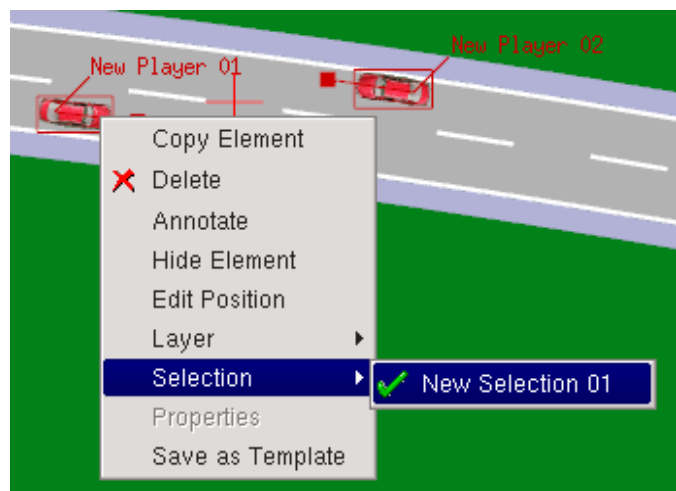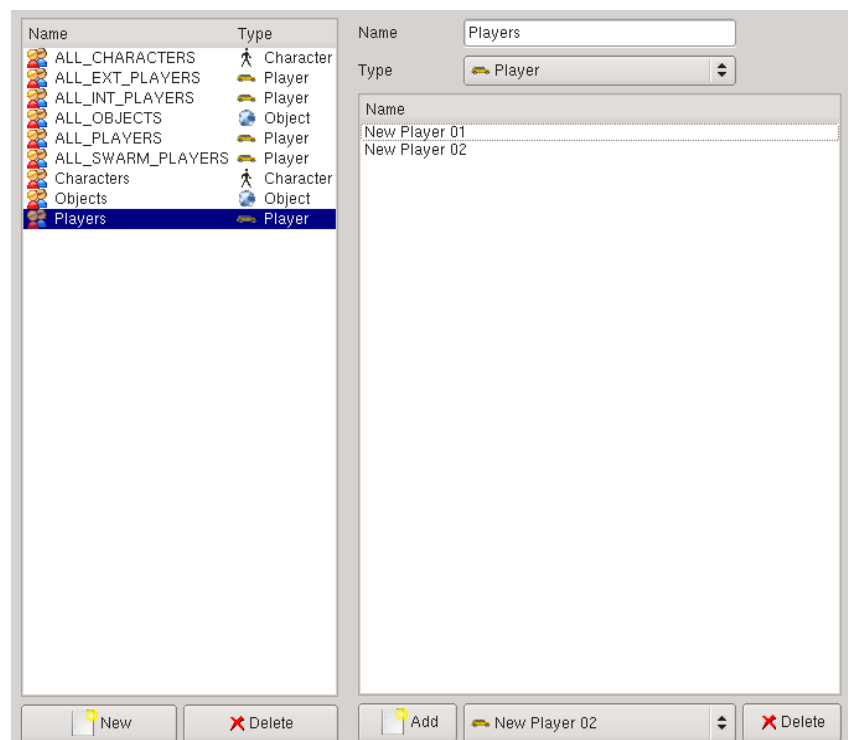| Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.:	Issue:		Page:
		VI2008.027	M		14 of 39

Copyright note according to DIN 34.

Within the **Layers** tab, new layers can be created or available layers can be deleted. Also the layers can be renamed.

Layers are very helpful for the work in complex scenarios. By assigning an element to a layer (right click on element to assign to a layer), the element is only visible if the layer is checked as visible in the Layers tab. In this way, elements like Players, Characters, Objects or Actions can be assigned to a specific layer in order to classify them for instance to specific situations.

The **Selections** tab provides functionality for grouping characters, objects or players. On the left side all selections are shown. With a double-click on the selection name in the list, the actions menu will open and new actions and triggers can be assigned to all members of the selection. The first selections in the list, starting with "ALL_...", are standard selections and therefore not editable. With the buttons below, new selections can be created or existing selections deleted.

On the right side user-defined the name and the type of the selection can be modified. By changing the type, all elements will be deleted from the list, to avoid selections with mixed elements of different type. This is necessary, because actions are type-specific. In the middle is also a list of all elements of the selection and below that, new elements can be added to or deleted from the selection.

This can also be achieved by right-clicking and opening the menu on an element and clicking on Selection. By selecting the selection in the sub menu, the element is added or deleted from the selection.

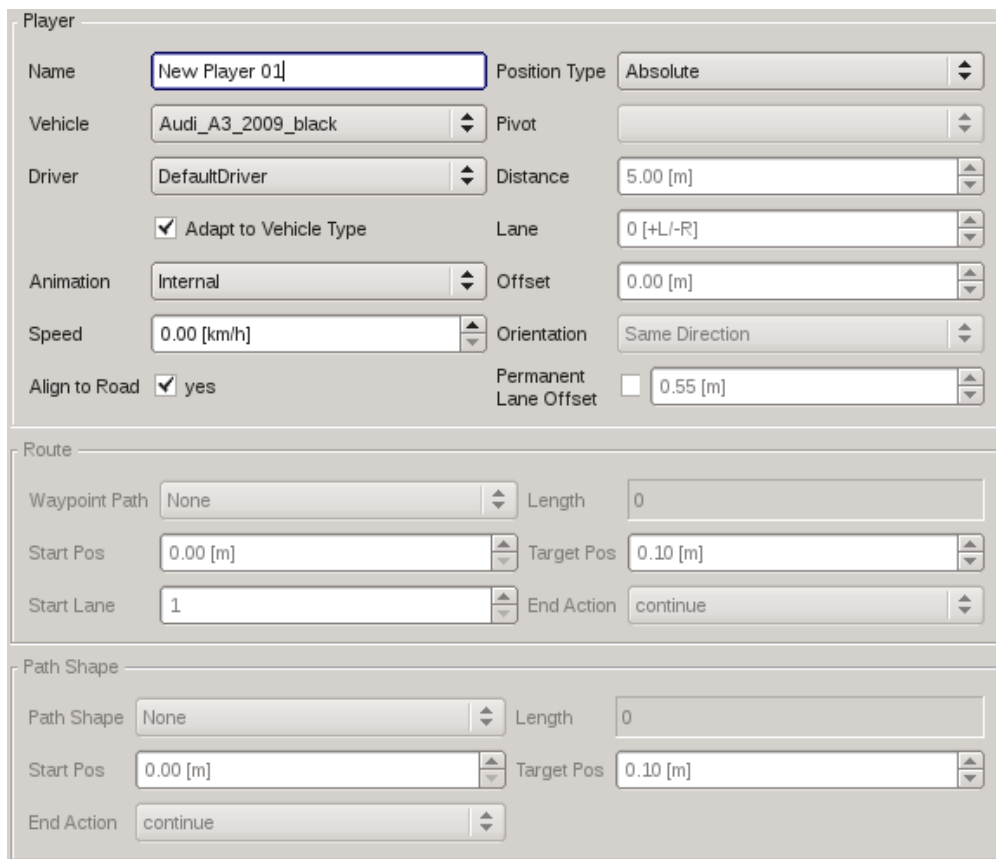| Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 15 of 39

All of the dialogs, like Vehicle Definition, Driver Definition, Traffic Elements, Counters and Layers have two buttons in common:
- Button *New* creates an entry with standard values.
- Button *Delete* removes the current selected entry.

Every vehicle definition and driver definition can be copied (button *Copy*) and reset (button *Reset*) to default values. If the entry is read from a database, the values are reset to the database settings.

2.3   Instantiating and Positioning Players

New players may be created and positioned in the scenario if the tool 🚗 (see left border of main window) is activated. A player is then added by clicking on any position of the road network. A double-click on the symbol 🚗 opens the *Player Properties* dialog (see figure below).

In the tab *Initialisation*, general settings of the player can be defined including the initial conditions that are valid upon start of the simulation as well as after a reset.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 16 of 39

Copyright note according to DIN 34.

The following properties are available:

Name	The unique name of the player. Upon creation of the player, the name will be „New Player", extended with a number increasing with each player.
Vehicle	The vehicle type of the player, which may be selected from a list showing all available types that have been defined for the player definitions (see chapter 2.2).
Driver	The driver type of the player that can be selected from the list of all previously defined drivers. It is always possible to assign "No Driver" to the player a special type of driver which is first in the list. In this case the player would virtually need no processor time, but is then not able to drive either action or autonomously controlled. This can be useful for parking cars.
	Using *Adapt to Vehicle Type* modifies parts of the driver behavior to achieve more realistic characteristics. E.g. adapting a DefaultDriver to a truck will result in a larger distance keeping, compared to a motorbike.
Animation	The type of player control. With this selection list, the user determines whether the player is calculated by the traffic simulation (internal) or by an external task or module. External players are shown in blue color in the editor while internal players are shown in red.
Speed	Defines the initial speed of the player.
Align to road	If activated, absolute positioned players are orientated exactly tangential to their starting road position directly at the start of the simulation.
Position Type	Allows for the selection in what way a player shall be positioned on the road network. Available position types are:
	Absolute coordinates cause the player to be positioned at exactly the inertial position where it is shown in the editor (provided that this position is on a road). In the main window, the user may vary the initial orientation of the player by means of a handle that is displayed at a selected player. Internally calculated players (see *Animation*) are automatically oriented in tangent direction of the street after start. Absolute players are highlighted in red color.
	PathShape coordinates enables the placing on a user-defined path shape. By selecting the path shape from the list, the element is placed on the start position of the path shape. The start position and also the target position can also be changed there. The *End Action* defines the subsequent action after reaching the target position. The player can be stopped, moved autonomously afterwards or drives the path again with loop, if the path shapes allows this action. The *PathShape* position type is similar to the *Route* position, only for path shapes.
	Relative coordinates allow for the positioning of the player relative to an existing player which is to be selected in the *Pivot* selection list. The *distance* to the player (measured along the road's center line) must be defined in the respective field. Positive values for the distance indicate that the player is positioned in front of the pivot player, negative ones behind. The lane position relative to the

pivot player can be defined in the *lane* text field. In addition, also the relative *offset* within the lane may be defined. With the *Orientation* drop down list, the user may define whether the player shall point into the same or opposite direction as the pivot player. Relative players are highlighted in magenta color.

If the player's start position shall be defined relative to an external player, please take into account, that the exact position of the external player may not be known before the start of the simulation. Therefore, when preparing a scenario, it is recommended to place the external player at approximately the position where it is expected to be upon start.

Route position allows placing a player on a predefined route that he must follow. When chosen, the route dialog which previously was disabled now gets enabled. The user can select a route from the list and is then displayed its length. If the current position of the player lies on the selected path, starting position and lane are determined from it. Nevertheless it is possible to define the start and target position within the route as well as the starting lane manually. *End Action* is an option that defines what to do when the end of a path is reached. The player can *stop* there, *loop* to the beginning or *continue* his travel randomly if possible.

Players positioned on a route are highlighted in orange color and cannot be moved with the mouse on the map unless a different position type is chosen.

Track position cause the player to be positioned in track coordinates. When the simulation is started, the player is automatically positioned on the defined road description.

Players positioned in track coordinates are highlighted in cyan color.

Trailer defines the position of a trailer behind a vehicle. By choosing the proper pivot element, the position of the trailer is changed by vehicle that is pulling the trailer. During simulation this player is only pulled and changed by the chosen pivot player.

Players positioned as trailers are highlighted in purple color and cannot be moved with the mouse on the map unless a different position type is chosen.

Position types *Absolute* and *Route* allow specifying a *Permanent Lane Offset*. This offset to the center of the lane will be kept by the driver during simulation.

Players can be deleted from the scenario by selecting them in the main window, invoking the context menu with a right-mouse click and selecting *Delete* or hitting the *Delete* key on the keyboard.

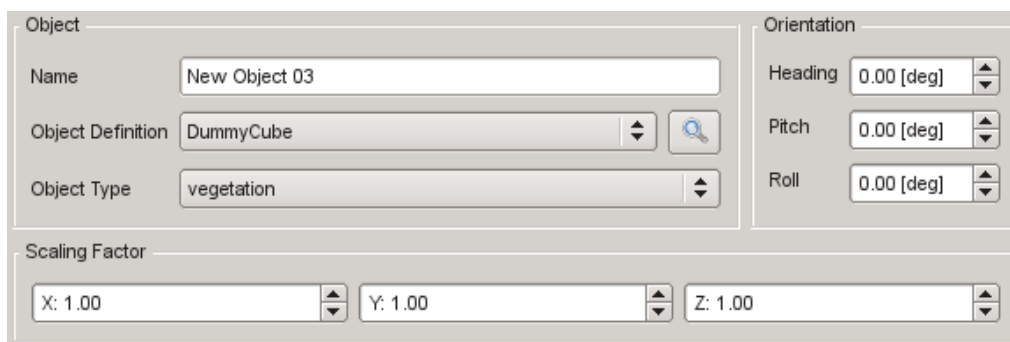## 2.4   Instantiating and Positioning Characters

When a new character was created and its symbol is double-clicked by the user, the Character *Properties* dialog is opened with tab *Initialisation* for characters. The figure shows an example how it could look like and differs with the available character types and models.

The following properties are available:

    Name   The unique name of the character. Upon creation of the character, the name will be "New Character", extended with an increasing number.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 18 of 39

Copyright note according to DIN 34.

Character Type   Defines the general type of the character and set of moves and gestures.

Model   Shows all available graphical models to the chosen character type.

The character types available to the user are read at program start from the file specified as CharacterModels -> descriptionFile defined in the Scenario Editor settings. (see Chapter 2.13). This file also lists the motions and gestures assigned to the character types. The user has to ensure the availability and accuracy of that file if he wants to use characters.

Via the button *Character Viewer* an external tool for having a preview at the characters and their motions can be launched.

2.5   Instantiating and Positioning Objects

When a new object was created and its symbol is double-clicked by the user, the Object *Properties* dialog is opened with tab *Initialisation* for objects. The figure shows an example how it could look like and differs with the available object types and models.

The following properties are available:

Name   The unique name of the object. Upon creation of the object, the name will be "New Object", extended with an increasing number.

Object Definition Defines the model of the object. 🔍 allows searching for object definitions by name and/or type.

Object Type   Defines the general type of the object.

The objects available to the user are read at program start from the file specified as ObjectModels -> descriptionFile defined in the Scenario Editor settings. (see Chapter 2.13). In addition, the orientation and the scaling can be changed.

2.6   Action-Control Concept

Per default, internally animated players are behaving autonomously, i.e. they will follow the road, obey the maximum or recommended speed and keep a reasonable distance to other players.

If a deterministic behavior of the player is required, then it needs to be provided with the applicable actions. These may be created either by changing to the tab *Actions* in the *Player Properties* dialog and pressing the button *New (element pos.) / New (cross-hair)* or by selecting a player in the main window, activating the symbol ⚠ and clicking onto a location in the road network. The first variant of the *New* button creates the   new trigger exactly at the current position of the player, the second variant creates the new trigger at the position of the cross-hair in the center of the map view.

Once a player owns at least one trigger for an action, it is considered to be action controlled. Autonomous behavior can be switched on again by means of the correspondent action.

In general, actions are linked to trigger points, meaning that certain conditions must be fulfilled in order for an action to become active. The list of all trigger points that are defined for a player is shown in the *Trigger List* (see *Actions* tab or the players *Properties* dialog).

Players that have no driver cannot own any actions, because this is a mode to save processing time for a player intended for e.g. huge amounts of parking players and therefore – among others – no evaluation of triggers can be done. Users will be warned at simulation start, if actions have been assigned to such a player.

2.6.1   Triggers

There are different types of triggers available:

- *Absolute:* The trigger is located in absolute (inertial) coordinates. It may be positioned with the mouse in the main window. The *Activation Radius* indicates the circular range within which the actions linked to the trigger will become active. It is indicated in the main window by means of a blue circle.

- *Track:* Actually, this is also an absolute trigger, however it is based on the road description. By indicating a *lane*, it is made sure that the trigger will only be activated by players passing it on this lane. This restriction may be overridden by selecting *any* lane for activation. Track-based triggers do not need an activation radius since the player's road position is used for the activation. The triggers are displayed with a cyan circle in the main window.

The two trigger types described above may also be assigned to players other than the one which is currently selected. Providing a *pivot* player that is different from the own entity causes the actions linked to the respective trigger points to be activated when the given player passes the trigger point. The actions themselves, however, will always influence the selected player. If no pivot player is given, the selected player will trigger itself.

- *Relative:* This trigger point is activated when the distance between the selected player and the *pivot* player is below the given *Activation Radius*. Any player may be selected as pivot player. The trigger point is assigned to the pivot player also for the graphical display and can not be moved with the mouse. Relative triggers are indicated in the main window by means of a magenta circle.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 20 of 39

Copyright note according to DIN 34.

- *External:* This trigger variant - visualized by a black circle - is not location based but depends on a command from another simulation module (e.g. the operator station). The trigger is identified by a user definable string. In case an external trigger is activated via network command all triggers referencing this external trigger id are considered to be active. An external trigger condition stays active until the corresponding deactivation command was received. The command format for a trigger called "myTrigger" is as follows:
  ```
  <Traffic> <Trigger id="myTrigger" active="true | false" /> </Traffic>
  ```

- *TTC Inertial:* Calculates the projected velocity of the pivot to the given position and deduces the time that it would last to move there with current constant speed. This trigger fires if this time decreases below the given TTC. The size of the dark blue circle around the trigger indirectly corresponds to the time to collision.

- *TTC Relative:* Calculated similar to TTC Inertial, but considers the relative velocity between action owner and pivot as well as pivot's current position. Triggers of type TTC Relative are marked by means of a dark magenta circle in a size representing the TTC.

- *TH Relative:* Considers the relative time headway from the action owner to the pivot. TH Relative triggers are indicated by means of a pink circle. The chosen time headway corresponds to the size of the circle.

- *Absolute Speed:* This trigger is activated if pivot exceeds (on excess checked) or falls short of (on excess not checked) the given activation speed. The trigger is indicated by a light red circle, whose radius is proportional to the activation speed.
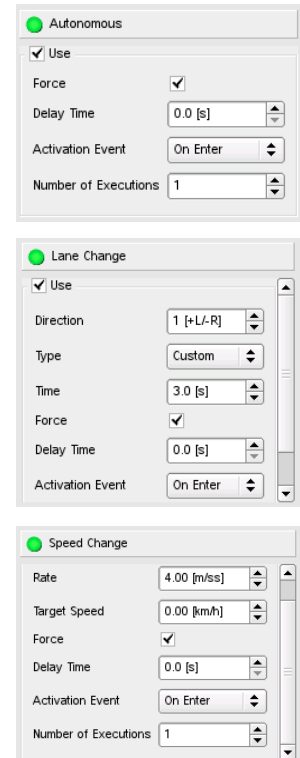
All triggers may also depend on so-called *Counters*. These counters are globally defined and may be referred to by selecting a counter in the field *Counter Id.* Secondary a comparison operator as well as a numerical value have to be provided which form a statement that has to become true before the trigger can get active. Counters are usually used to trigger different actions if a player passes a point in the database multiple times (e.g. in endless racing courses).

2.6.2 Player Actions

For players exclusively, these different actions are available, some of which may also be active concurrently:

- *Autonomous*
  This action sets the player to autonomous mode until a new action becomes active. It may only interrupt running lane- and speed change actions if the *force* checkmark is set, otherwise the action would have no effect.

- *Lane Change*
  The player will change the lane. The direction and number of lanes can be defined in the field *Direction*. Positive numbers indicate a change to the left, negative ones to the right. The duration of the lane change will be determined by the data given in the fields *Type* and *Time*. Only a single lane change action may be active at any given time (i.e. a second action will be ignored). To cancel a potential running action of this type, the *force* flag can be set.

- *Speed Change*
  The player will change his speed to the given *Target Speed* with the given *Rate*. Only a single speed change action may be active at any given time (i.e. a second action will be ignored). To cancel a potential running action of this type, the *force* flag can be set.

All player and character actions have three attributes in common:
- *Delay Time* specifies the time in seconds by which the activation of the given action is delayed after the trigger condition was fulfilled.
- *Activation Event* states if the action should be started *on enter* (conditions are met) or *on exit* (conditions are false again)
- *Number of Executions* defines how many times the action will be started altogether in one scenario run. Available options are positive numbers, *infinite* and *never*.

If actions are assigned to an external player, that is not controlled by the traffic module at all, the module controlling this player can be informed via network about the actions becoming active. In this case the external module completely takes responsibility for reacting to those actions.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 22 of 39

Copyright note according to DIN 34.

2.6.3 Character Actions

For characters exclusively, these different actions are available, which typically may also be active concurrently:

- *Motion*
  This action activates a motion that shall be executed by the character. All different motions that are available are shown in the *Move* list with the average speed of this motion written in brackets. A value of "<automatic>" tells the module to find the best motion for a given speed. For all other motions *Motion Speed* tunes the move to be the desired speed, a negative value takes the - usually best looking - default. If *Force Move* is selected, the new motion is started immediately without trying to blend over.
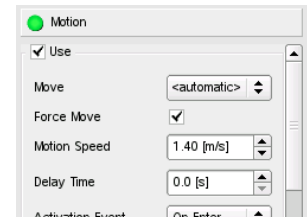
- *Path Shape*
  With this action the character can be set to the beginning of any predefined path shape (except clothoid shapes) shown in the *Path Shape Name* list. One and the same shape can be used by multiple characters. The option *Beam* directly transfers the character to the start of the shape, when the action is activated. Without "beaming" the character first will automatically create a bridge path as shortest connection from his current position to the start of the shape and thereafter follow the path shape.
 If *Loop Path* is selected a moving character reaching the end of his path will continue from the beginning again. Otherwise he would stop at the last waypoint of his path.
  With *Clamp To Ground* set active, traffic tries to continuously keep the character on the ground (based on the given XODR-file). Reasonable z-coordinates in the given path shape are required for this function to work correctly.
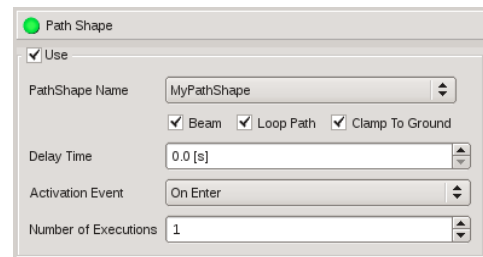
- *Gesture*
  Gestures are moves that can be executed while performing any 'normal' motion e.g. waving, shaking head. All available gestures for the character type are listed under *Gesture Type*. The number of repetitions can be specified with *Repeat Times*.
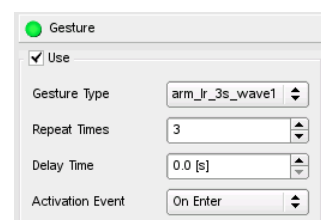
Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 23 of 39

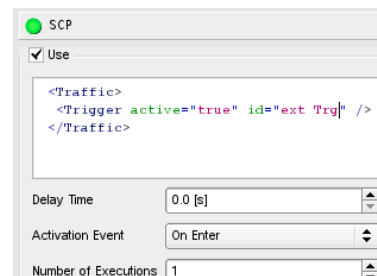Copyright note according to DIN 34.

2.6.4 Common Actions

These actions are available both for players and for characters because they represent scenario global influence:

- *SCP*

  This action type makes it possible to emit scenario control commands that can influence the traffic module, but can also have any other simulation participant as receiver. This is what it is intended for. The user can define own messages in XML style as shown in the example here.

  General syntax errors will be signaled by a red frame around the text edit area. An additional tooltip may provide more detailed information about the error.

  Within SCP messages instead of using the name for a player, character or object the wildcard $owner can be used. The traffic module simply replaces this wildcard with the owner of the action. A more sophisticated way to assign an action to an element is to use the wildcard <player>$pos<pos>:<lane>, where <player> is the name of a player, <pos> is the relative position and <lane> the relative lane to the player. For example Ego$pos1:1 is the next car on the right lane relative to Ego and Ego$pos:-2:-1 is the second player behind Ego on the left lane.
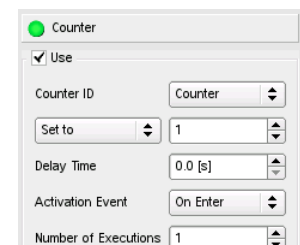
  Besides typing the messages manually a more user-friendly approach is to use a generic GUI that is launched automatically from this action's dialog or can be opened via the *view->SCP GUI* menu entry to be used for composing the messages. Section 2.13.2 explains the configuration of the GUI. The structure of the messages to be used with the GUI has to be: *group* (with optional attributes) -> *command* -> *attributes*
  e.g. `Traffic` -> `Trigger` -> `active, id`

- *Counter*

  The global counter given in the field *Counter ID* will be modified as defined when this action is triggered. However, it will only be triggered if all conditions associated with the action's trigger are fulfilled. If, for example, the trigger of the counter action depends on another counter, it can only be activated it the latter has the correct value. Supported modifiers for counters are *increasing by*, *decreasing by* and *setting to* certain values.

Date: 15.01.2016	Title: User Manual			
Name: Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 24 of 39

Copyright note according to DIN 34.

2.6.5 Selections

A selection describes a group of players, objects or characters that can trigger a player's action or perform a certain action. A player, character or object can be member of multiple selections at the same time.

Whenever a selection can be selected or is displayed in the editor the icon ![icon] is used.

The following predefined selections exist:

ALL_CHARACTERS	every character is member of this selection
ALL_EXT_PLAYERS	the group of external players
ALL_INT_PLAYERS	the group of internal players
ALL_OBJECTS	the group of all objects
ALL_PLAYERS	external and internal players
ALL_SWARM_PLAYERS	internal players that belong to a swarm

2.6.5.1 Managing Selections

The selections can be managed within the properties in the selections section (see Preparation of Scenarios). User-defined selections can enhance predefined selections, by assigning only certain players, characters or objects to a selection group. It is also possible to add or delete a single element by right-click on the element.

2.6.5.2 Selections as pivot

A single player who owns an action can use another player just like himself as pivot for the trigger of his actions. There is only the exception of external triggers, because in this case pivots cannot be used at all in general.
Additionally the user can choose a selection as pivot for all not external triggers. This means that every single member of the selection creates enter and exit events for this trigger as long as the last referencing action was not finally terminated.

An action, triggered by a selection, can be executed multiple times. It is possible that different members of the selection trigger this action. In the other case the first one meeting the trigger criteria fires the trigger.

2.6.5.3 Selections as owner of triggers and actions

It is possible to define a trigger with actions once and assign it to a complete selection. This means that every member of the selection will be capable of running its own copy of this action when triggered.

To assign a trigger to a selection first it has to be created as described in 2.6. After that, by right clicking on the trigger item in the trigger list, a menu opens and the trigger can be moved via the *Transfer To* option to the desired selection.
This trigger list context menu can always be used when the user wants to copy or move actions between elements or selections.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 25 of 39

Copyright note according to DIN 34.

Triggers that belong to any selection are painted with a greenish icon in the map in contrast to regular triggers that are marked with yellow icons.

The trigger list of every player always shows the triggers of player selections. The same applies to characters and objects. This results due to the fact that selections have no physical representation besides the players / characters they are formed of.

With reference to pivots the following has to be considered:
- Any element can be pivot of a selection's trigger. This means exactly this element would trigger the actions of the complete selection.
- Selection as pivot: Just like for single players all members of the selection can potentially trigger any selection's action.
- <owner>: this is a wildcard, that only appears in the pivot list for selections. It means, that although a group of players owns this action (or more precisely a copy of it), in each case only the respective owner will be considered as pivot for triggering.
  When using SCP-Actions, that refer to players, in conjunction with selections the $owner wildcard can be used in place of the regular player name for assigning the SCP command to every player of the selection.

2.7   Scenario Templates

Scenario Templates is a mechanism to export certain parts of a scenario in an own template file in order to import the saved situation to another scenario. It is possible to export elements like characters, objects, players and actions.

For example a VDA lane change with pylons that are exactly positioned, can be exported to an extra file and then imported to any other scenario.

Other situations can include traffic-jams, accidents and so on. The functionality will be extended in further versions.

2.8   Measuring Tool

By right-click on the map and selecting *Measure – Add Measure Point* an arbitrary point in the map can be added to start the measurement. By adding further measure points distances and angles can be easily measured. The points can be moved with the mouse. In addition, all measure points can be deleted by the right-click menu point *Measure – Remove Measure Points.*
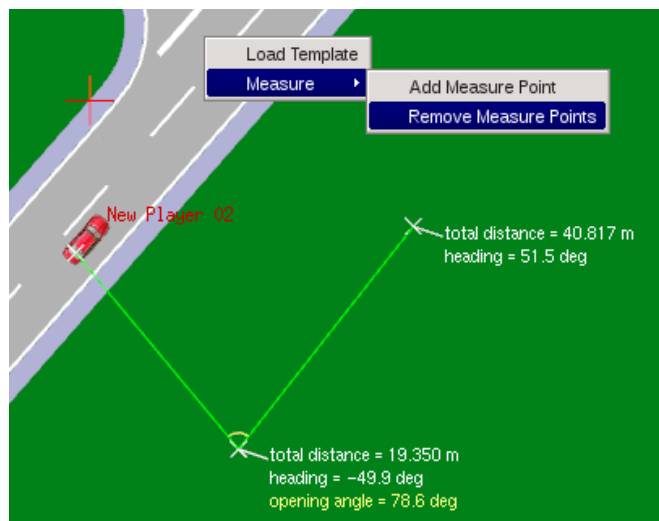
Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 26 of 39

Copyright note according to DIN 34.

2.9 Configuring Traffic Lights

To understand how the configuration works it is best to first explain the concept that stands behind. Two main types concerning traffic light definition are distinguished:

- **Controllers**: they are defined in the road description by the creator of the database and combine an arbitrary number of traffic lights that share the same switching states simultaneously during runtime. That is the reason for defining the phases (the logical meaning of what a light sign shows) at the controller layer. Controllers have no physical representation itself in the visual database.

- **Traffic Lights**: they are also defined in the road description as a concrete signal visible in the database. A traffic light must always be referenced by exactly one controller. The user can define the light states that are switched with the controller states. Different traffic lights sharing one controller can have completely different light states for the same phase.

The dialog to configure the traffic lights including phases and light colors can be opened by either pressing the *Traffic Lights* icon, choosing the action from the *view* menu or pressing the assigned keyboard shortcut.

2.9.1 Tools of the Traffic Light Dialog

If this toggle button is activated the view jumps to the selected traffic light in the map.

This toggle button expands the tree to show all traffic lights belonging to the controllers.

the tool to copy the definition of a controller or a light sign, shortcut *Ctrl+C*

the tool to paste an afore copied definition of a controller or light sign, shortcut *Ctrl+V*
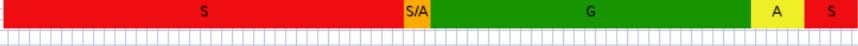
Starts a process to check all definitions for those that are not defined in the road description and can be automatically removed. This is only done on user request, because it is possible that the user loaded the wrong layout file that does not suit the traffic light definitions, but no already made configuration shall be lost. Configurations for controllers or lights that do not exist in the road description are ignored by the traffic module.

Filter Controllers can be used to limit the display of controllers to controllers of a selected junction only.
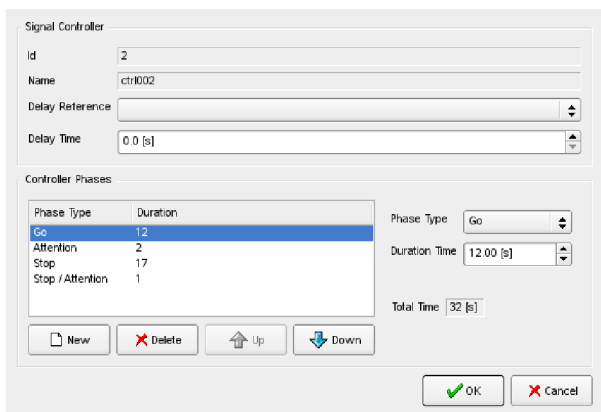
2.9.2 Editing Controllers

When the dialog first opens, all available controllers are presented to the user in a list. Available means either defined in the road description and / or there have been phase definitions or light state definitions found in the currently loaded scenario. Should a controller or a traffic light only be referenced by the current scenario, the column *Def. In Road* will indicate this with entry value "*no*". The width of the colored phases corresponds to their time duration. To scale the view the slider on top of the dialog can be used; scaling has no influence on the times.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 27 of 39

Copyright note according to DIN 34.

If no phases have been created so far, the second column shows a *Create Phases* button which opens the edit dialog. If there are phases defined - like shown in the picture - the dialog opens by clicking on the phase you want to edit pre-selecting it. It is also possible to double click on the row to open the dialog.

The dialog shows id and name of the controller as specified in the road description. The drop-down list *Delay Reference* shows all other controllers where one can be chosen to have a defined delay to. One controller having a delay to another one means that its first phase virtually starts *delay time* seconds after the start of the reference's first phase. This can be used to define a progressive signal system, but only makes sense, if the total times of all connected controllers are the same.

The list at the bottom of the dialog shows all currently defined phases and provides the tools to create a new, delete or move a phase in the list. Phase types are important for the behavior of the traffic, they do not describe which lights are visible, but what the light signal means at a certain time. Available are:

- off   traffic light is off and has no meaning
- stop   traffic must stop at the light
- stop / attention   traffic stops, but soon will be allowed to go again
-
- go   traffic may drive, but has to attend right of way rules (oncoming traffic)
- go exclusive   traffic can go and does not have to wait for others (turn with separate arrow)
- blink   the light state is switched on and off repeatedly, otherwise like state "off"

2.9.3 Editing Light Signs

If a controller is expanded, it shows its assigned traffic lights.

In this case we would have a controller with id 308 and one assigned traffic light with id 21897. As the phases are already defined, the suitable light states need to be defined too. If no state is defined a *Create Lights* button is shown, otherwise the user can select one light state or double click the row to open the lights edit dialog.

It shows id and name of the light sign as defined in the road description, as well as the controller it belongs to. The list shows all currently defined phases, that can be managed. Buttons to add, delete and move light masks are provided.

The amount of lights and the available colors are defined in the corresponding dynamic objects configuration file for the signals type/subtype combination.

The selected mask in the example shows an active yellow light for an "attention" controller phase.

2.10 Action Info Monitor

This dialog keeps track of all actions started and finished during the run of a simulation. In a table with sortable columns the user can easily monitor when (SimTime) an action started and read who the owner is, which ID and type the action has and what the name of its trigger is.

When an action is finished the icon turns from green to red and the exit simulation time is written in brackets.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 29 of 39

Copyright note according to DIN 34.

If actions have additional parameters like time for lane change, those will be presented in the *parameter / value* window for all selected actions in the table.

The table is only cleared on the user's request by pressing the *Clear* button.

To disable the sending of Action Infos by the traffic module it can be started with option `-noInfo`. During runtime to following SCP command can switch on and off action notification:
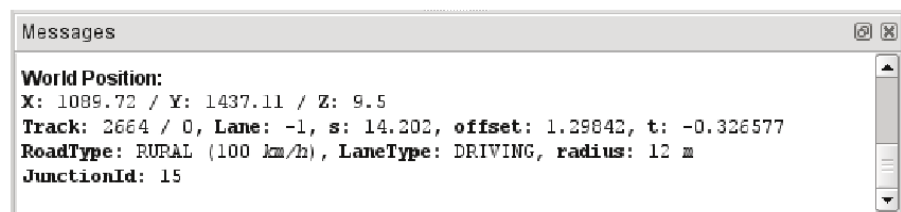`<Traffic><Config actionNotice="true/false"/></Traffic>`
The dialog shown on the side can be opened with the help of the *Owner Filter* button. There all players for which action notifications have been received are listed. The user may select only the players that he is interested in for displaying in the table.

Check All can be used to switch on and off all check boxes at once.

2.11 Message Dock Window

All messages created by the editor that are of interest for the user or help finding errors are written to the message dock window. This window can be shown and hidden over the
view menu or the assigned keyboard shortcut. It can be moved to bottom or top of the main window or completely released. It is possible to copy selected text and to completely clear the message log by selecting *clear* from its context menu.
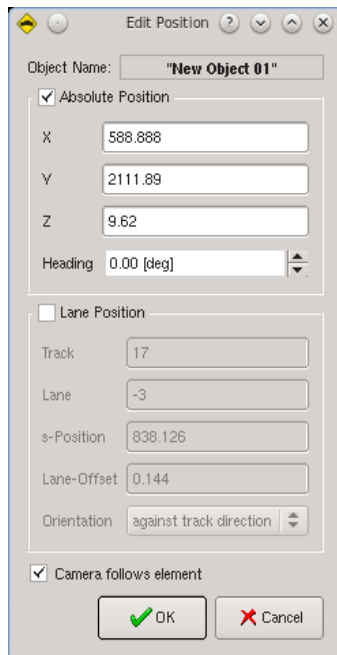
Note: error messages resulting from reading the initialization file, concerning license errors or any other errors that prevent the editor from starting successfully are only printed to standard out / error.

The message dock is also used to show lane picking information created by pressing down the *Home* button on the keyboard and *right-clicking* to any position on the map. Depending on where the user clicked more or less information can be retrieved from the road, like e.g. membership to a junction.

2.12 Editing Element positions

Besides moving elements like players, characters and actions with the mouse, an exact position can be entered via the *Edit Position* dialog. It is opened if an element is right clicked and Edit Position is chosen from the context menu.

Date: 15.01.2016		Title: User Manual		
Name: Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 30 of 39

Copyright note according to DIN 34.

The name of the element is shown at the top.

In the upper half of the dialog the absolute position in the map is shown. Depending on the type of element the input field for the z-coordinate may be disabled. The input fields are filled with the current absolute position per default. The user can change it and press *OK* to save the changes.

In the lower half of the dialog the lane position of the element is shown if it could be determined. To move the element by means of a lane position the Lane Position checkbox has to be activated.

The position is tested when the user pressed the *OK* button. Only valid positions are accepted, in the other case the user is shown a message and the element is not moved.
If an element was successfully moved the view on the map moves with it.
Players that are positioned on a route or as a trailer and actions that are relative to another element can not be moved using this dialog.
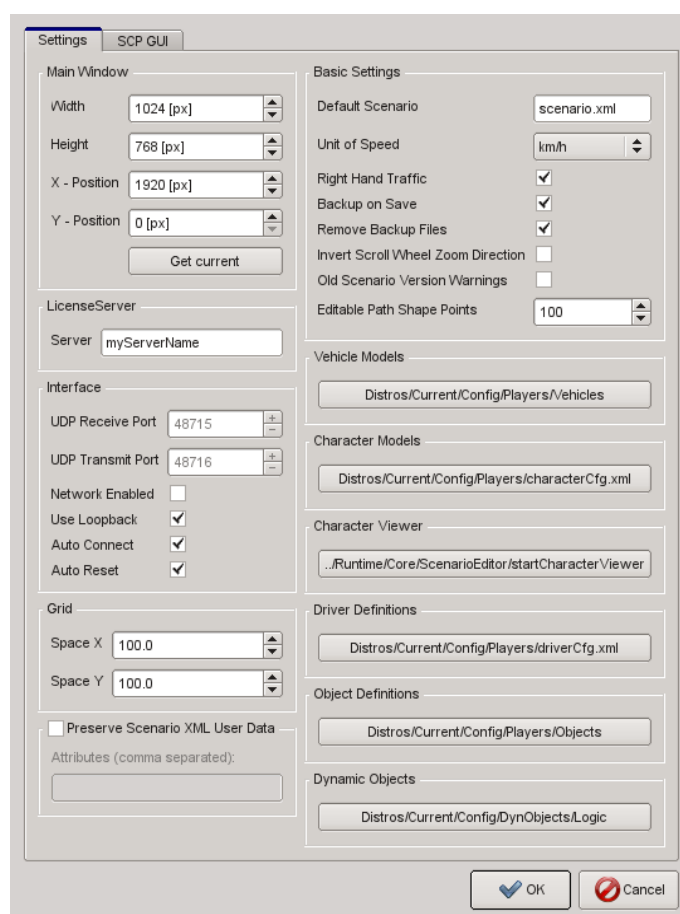
2.13 Scenario Editor Settings

Via its configuration file the Scenario Editor can be adjusted to the users needs for instance in case of window size and position at start-up, grid dimensions etc. Some other features have to be defined in order to connect the editor to other modules like image generators or task control. Although the file format is XML, it does not have to be edited manually, because the editor provides dialog *Editor Settings* under menu *Edit* to conveniently set all features. The following sections explain all settings that can be defined. After changes have been made to the dialog, the new configuration file is written to disk. For most settings the Scenario Editor has to be restarted in order to reread the file. Please see also the tool tips help of the dialog.

2.13.1 Basic Properties

name	meaning
Width	width of window
Height	height of window
X-Position	X-Position on screen
Y-Position	Y-Position on screen
Server	name or IP address of license server
UDP Receive Port	UDP receive port number
UDP Transmit Port	UDP transmit port number
Network Enabled	tells if network mode is enabled at start-up

Use Loopback	uses local loopback network device if true
Auto Connect	tells if auto-connect mode is enabled as default
Auto Reset	tells if auto-reset is checked as default
Space X	grid x dimension (works immediately after OK)
Space Y	grid y dimension (works immediately after OK)
Default Scenario	scenario to load after start if none is given from command line
Unit of Speed	use km/h, m/s, mi/h globally
Right Hand Traffic	tells if configured for right hand traffic
Backup on Save	create a backup file when saving a scenario
Remove Backup Files	remove all backup files created during current session
Invert Scroll Wheel	inverts the wheel scroll direction for the zoom
Old Scenario Version Warnings	show warning, if loading an old scenario
Editable Path Shape Points	Path Shapes with more points are displayed only (not editable)
Vehicle Models	shows and sets referenced vehicle database
Character Models	shows and sets referenced character database
Character Viewer	shows and sets external character viewer executable
Driver Definitions	shows and sets referenced driver database
Object Definitions	shows and sets referenced object database
Dynamic Objects	shows and sets referenced dynamic object configuration

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 32 of 39

Copyright note according to DIN 34.

2.13.2 SCP GUI

The definition of the generic GUI for SCP messages can be done in XML using the tab *SCP GUI*. All group entries and their child nodes have to stand within the enclosing `<SCPGUI>` brackets. The editor is capable of syntax highlighting and will try to find simple syntax errors. If the *OK* button was pressed and a parsing error was identified, a small window pops-up showing hints to the error including line and column numbers. The text editor has a context menu accessible through the right mouse button providing basic editor functionality like undo, redo, cut, copy and paste.

```
<SCPGUI>
    <Group title="Traffic">
        <Command title="Trigger">
            <Argument title="id"/>
            <Argument title="active">
                <Option title="true"/>
                <Option title="false"/>
            </Argument>
        </Command>
        <Command title="ResetPlayer">
            <Argument title="id"/>
            <Argument title="name"/>
        </Command>
        <Command title="Pulk">
            <Argument title="id" default="1"/>
            <Argument title="enable">
                <Option title="true"/>
                <Option title="false"/>
            </Argument>
            <Argument title="desiredSpeed" default="-1.0"/>
            <Argument title="centralPlayer" default="Ego"/>
            <Argument title="visibleCount" default="-1"/>
            <Argument title="semiMajorAxis" default="400"/>
            <Argument title="semiMinorAxis" default="400"/>
            <Argument title="offset" default="100"/>
            <Argument title="percentageFront" default="0.4"/>
            <Argument title="percentageBack" default="0.3"/>
            <Argument title="percentageLeft" default="0.15"/>
            <Argument title="percentageRight" default="0.15"/>
            <Argument title="percentageOwnSide" default="0.6"/>
        </Command>
        <Command title="Config">
            <Argument title="actionNotice">
                <Option title="true"/>
                <Option title="false"/>
            </Argument>
```

Line: 1 Col: 1

Below is an example derived from the external trigger definition with an additional fictional argument for the group. Keywords of the messages always are values of the corresponding title attributes.

```
<SCPGUI>
    <Group title="Traffic">
        <Argument title="group_argument" default="something" />
        <Command title="Trigger">
            <Argument title="id" />
            <Argument title="active" default="true">
                <Option title="true" />
                <Option title="false" />
            </Argument>
        </Command>
    </Group>
</SCPGUI>
```

There is no limit for the number of groups or commands. Multiple groups can have the same title, but inside a given group no two commands with the same name are allowed. Inside a command of a group no two arguments with the same title are allowed.

An argument will usually be shown as an input box in the GUI. If a default string is given it will be displayed there. The user can specify options for arguments that usually will be displayed as drop down lists. If a default for a drop down list is given this entry (if available) will be selected first.

All groups, commands and arguments appear in the order they were written in the configuration tab. If a title for a tag is missing it will be marked with "<noTitle>" in the GUI.

2.14 Map Tiles

The map in the center of the editor showing all roads can be extended by textures rendered from the visual database to facilitate orientation in the database for users. Those textures will be drawn behind any other visual elements.
First it is necessary to create the renderings as rectangular textures of identical size in the PNG format. Each texture may e.g. represent one square kilometer. At places, where there is no database defined, no textures should be created to save memory. The naming convention for the image files is as follows: `<anyNameWithoutUnderscore>_<horizontalIndex>_<verticalIndex>.png`

All textures have to be placed in one directory containing a file named *info* that contains the information about the physical size of the tiles in meters and the origin corresponding to index 0/0.
e.g.: `<MapTiles originX="-4000" originY="-2000" dimX="1000" dimY="1000" />`

When the editor is started with option *–tiles* plus the directory of the textures, it will read all textures providing textual progress feedback at standard out.
Please note, that in the map the textured background will only be visible underneath a certain zoom level.

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 34 of 39

Copyright note according to DIN 34.

3   Tips and Tricks

Get road details via shortcut *Home + Right Mouse:*
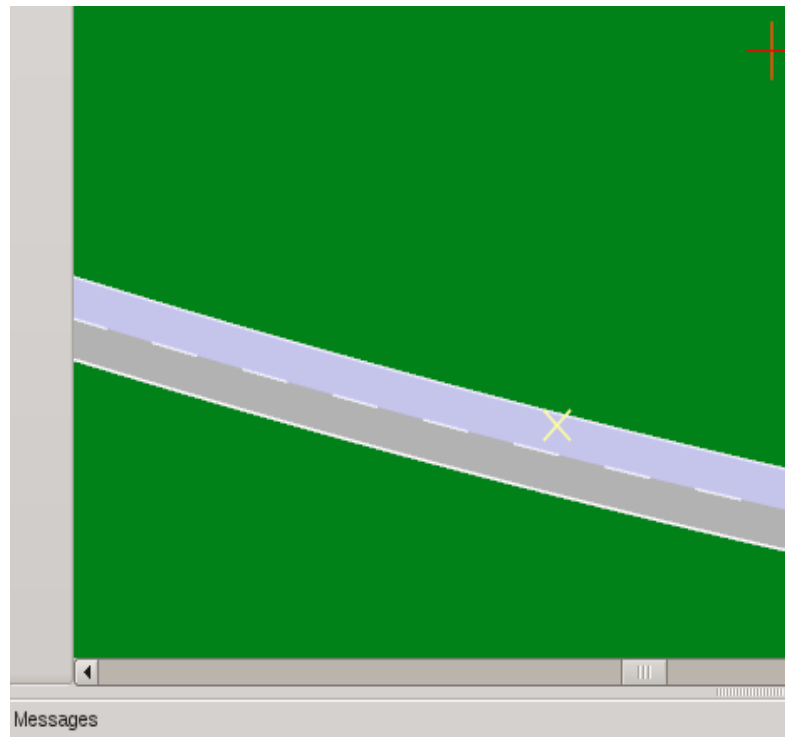
World Position:
X: 1583.07 / Y: 2356.42 / Z: -0.077675
Track: 0 / 0, Lane: 1, s: 1760.96, offset: 0.0684721, t: 1.94347
RoadType: RURAL (100 *km/h*), LaneType: DRIVING, radius: 1200 *m*

Get 3D-view via shortcut @:

Click for a longer time on overlapping elements to get a selection menu:

4   Shortcuts

File:

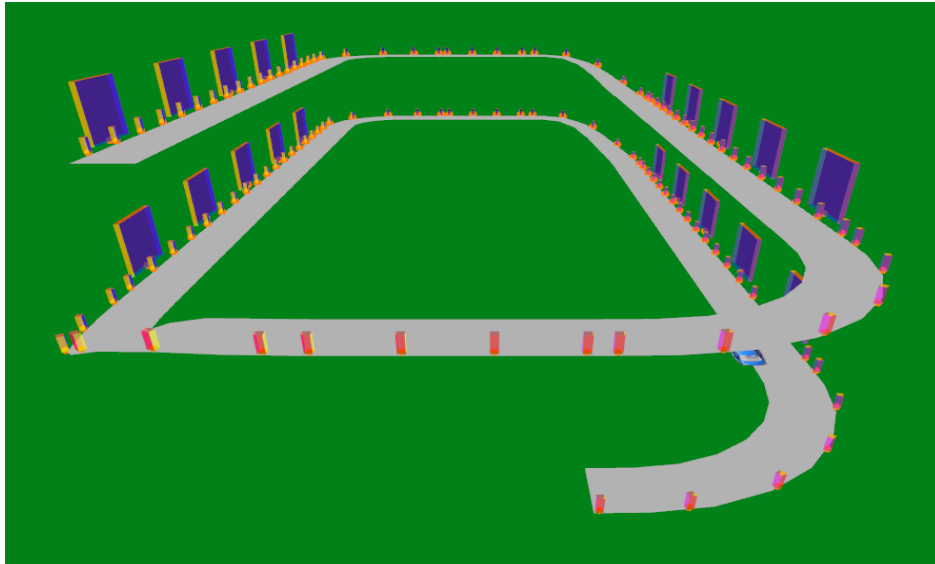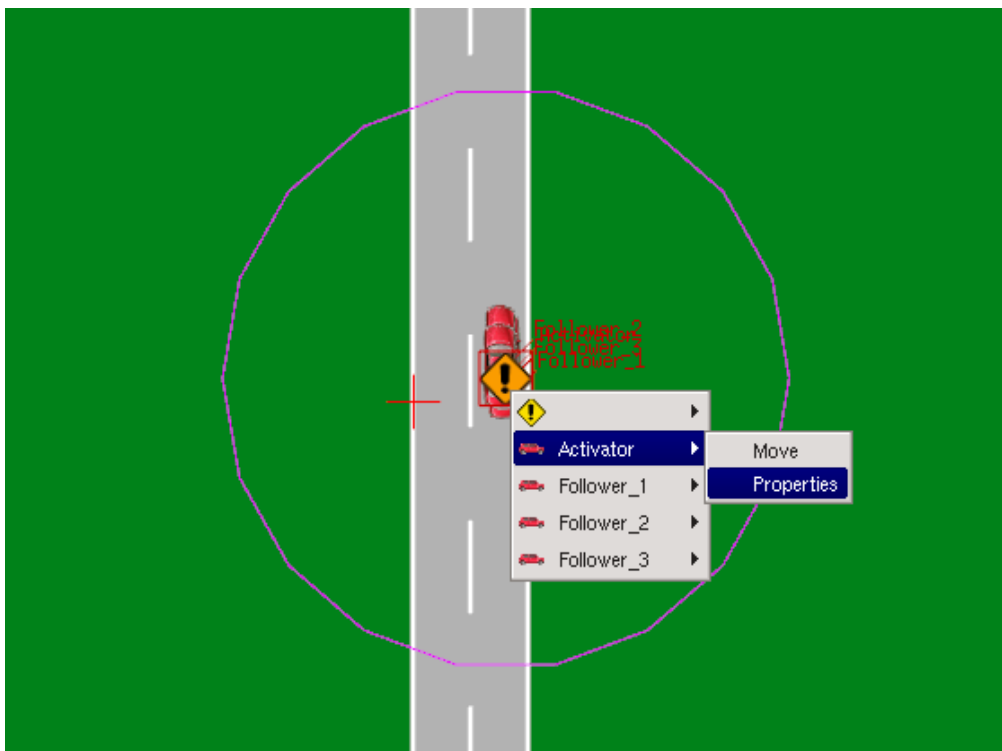|	
New File	Ctrl + N
Open File	Ctrl + O
Save File	Ctrl + S
Save As	Ctrl + Shift + S
Reset	Ctrl + R
Recover from Backup	Ctrl + B
Quit	Ctrl + Q

Edit:

Undo	Ctrl + Z
Redo	Ctrl + Shift + Z
Scenario Properties	Ctrl + P
Traffic Lights	Ctrl + L

View:

Road Signs	S
Traffic Lights	A
Sign References	R
Player Names	N
Vehicle Lights	L
Pulk Traffic Region	P
Database Grid	G
Database Origin	O
Border Lanes	B
Draw Database	D
Road Objects	J
Annotations	I
Path	W
Path Shape	E
Hide Selected Element	Ctrl + H
Show All Elements	Ctrl + Shift + H
Center View	C
Main Toolbar	T
Messages	M
SCP GUI	Shift + G
Player Heading Up	H

Extra:

Transform Elements	Ctrl + T

Network:

Network On	Ctrl + Shift + N
Network Off	Ctrl + Shift + O
Auto Connect	Ctrl + Shift + A
Auto Reset	Ctrl + Shift + R

Map:

Select	F1
Add Player	F2

Add Character	F3
Add Object	F4
Add Action	F5
Create Path	F6
Create Path Shape	F7
Show Scenario Monitor	Ctrl + Shift + M
Show Action Info Monitor	Ctrl + Shift + I
Zoom In	+
	Shift + Middle mouse + Drag down
Zoom In (to cursor)	Mouse wheel down
Zoom Out	-
	Shift + middle mouse drag up
Zoom Out (from cursor)	Mouse wheel up
Zoom 1:1	1
Selection Zoom	Ctrl + Middle mouse
Select Elements	Left mouse (+ mouse move)
(De-)Select Elements	Ctrl + left mouse
Move Left	Arrow left
Move Right	Arrow right
Move Up	Arrow up
Move Down	Arrow down
Move View	Middle mouse + Drag around
Rotate Elements	Space + Left Mouse
Rotate Elements Heading	Space + Right Mouse
Delete Object	Selection of objects + Delete
World Position	Home + Right mouse
Toggle Full Screen	Ctrl + F
Toggle 3D-Perspective	@

Traffic Lights:

Jump to selected	J
Show / Hide	A
Copy Setup	Ctrl + C
Paste Setup	Ctrl + V
Delete inexistent	Ctrl + Delete

Edit Traffic Light / Controller Phases:

New	Ctrl + N
Delete	Delete

Properties:

Vehicle Definitions:

New	Ctrl + N
Create Copy	Ctrl + C
Reset	Ctrl + R
Delete	Delete

Driver Definitions:

New	Ctrl + N
Create Copy	Ctrl + C
Reset	Ctrl + R

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.: VI2008.027	Issue: M		Page: 38 of 39

Delete   Delete

Traffic Elements:
New   Ctrl + N
Delete   Delete

Counters:
New   Ctrl + N
Delete   Delete

Action Tab:
New Action (element pos.)   Ctrl + N
New Action (crosshair pos.   Ctrl + Shift + N)
Delete Action   Delete

Messages Dock Window:
Zoom Text   Ctrl + Wheel
Clear Text   Ctrl + Delete

Date:	15.01.2016	Title:	User Manual		
Name:	Marius Dupuis et. al.	Document No.:	Issue:		Page:
		VI2008.027	M		39 of 39