Daga Games Studio

iTS – Intelligent Traffic System

Developer Manual

Jose Garrido
11/4/2014

Contents

Introduction

Intelligent Traffic System is an application developed for Unity3D that simulates traffic, this system controls cars in order to simulate traffic on a defined area.

Features included on this version

V1.1

Support for open lanes (you must enable this feature on the settings of the iTSManager - option name is "Allow dead end lanes?") the cars when they get into a dead end lane would get automatically removed and respawned on another location. (Needs reprocessing the road data)

There is now a script that would try to get the players position (lane, connector and point) so you could get a reference of where the player is, you must know that if the player does some really wacky behaviors, like crossing the streets in a totally uncommon way, the script may not be able to follow).

Fixed some behaviors that made, in some situations, cars not stopping correctly on junctions.

Improved the flow of the traffic on the junctions, trying to make it look more natural.

When a lane is deleted all traffic lights get updated, so your current traffic lights settings (connectors selected) won't get messed up.

You can set now a radius on the spawner to avoid initial spawning cars to get spawned where the spawner is, this is i.e. to avoid cars to be spawned where the player is.

You can also have the spawner check a list of transforms positions and its own defined radius to avoid cars getting spawned on top of them, usually useful if you have one spawner with an spawning area that covers the entire road map and have several players on the scene that you don't want cars to get spawned on top of them.

V1.0.9

Fixed a bug that sometimes a null reference exception was thrown when processing the junctions.

V1.0.8

Fixed a bug that made some cars wait forever on a junction.

V1.0.7

Added traffic cars to a parent object on the scene

Improved the pass priority behavior of the cars

Forced stops now makes cars to always do this stop first before moving on.

Connectors are released earlier now, which gives a more fluent flow of the traffic cars.

Added 2 variables to make the Length between cars random, to make it look more realistic

Improved the Lane occupation calculation

Performance improvements

V1.0.6

Improved performance of the lanes editor tool

V1.0.1

Added new feature to swap the lane direction while in edit mode.

Added the ability to add new points to lane and connectors at the end, while pressing the Shift key.

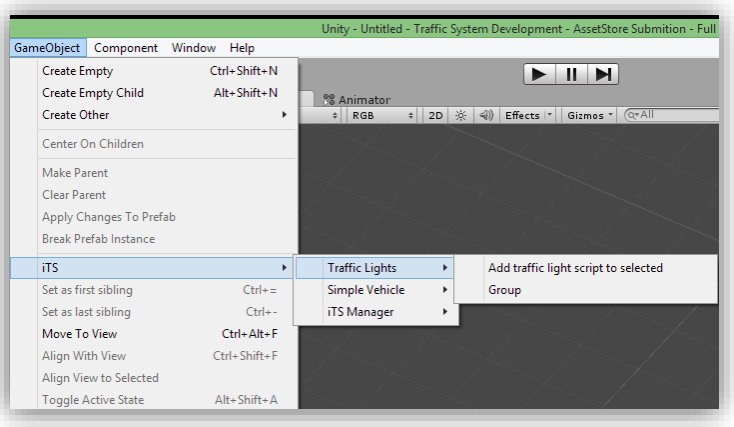
Added an int field to manually select lanes and connectors and to delete them also.

Contact Information

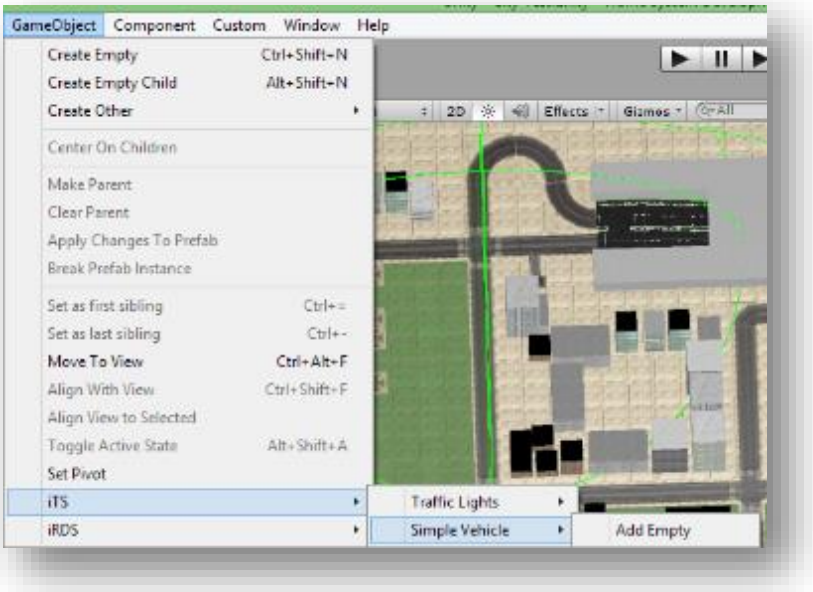
You can contact me by email on josegarrido@dagagames.com or by the unity forum user rhodnius with a PM.

iTS (Unity Menu options)

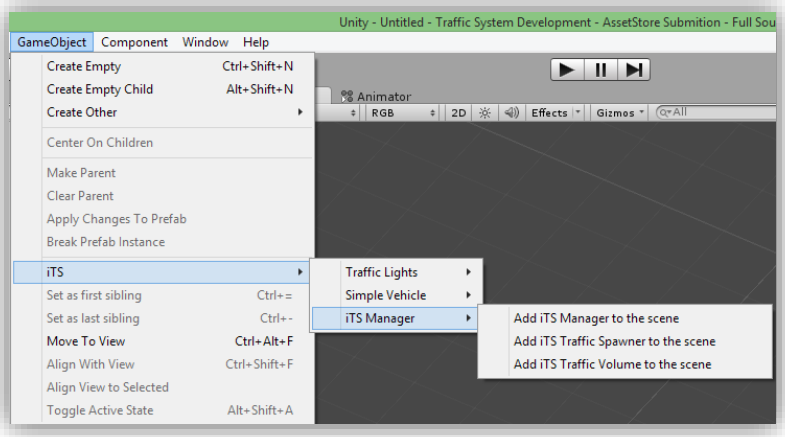
There are three options available when you import IRDS package into your project:



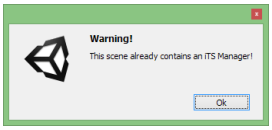
The first option (Traffic Lights) have the sub options Add Traffic light script to selected and Group, the first option allows you to add the traffic light script to a selected game object and the second option allows to make a group of traffic lights to enable an easier traffic light manager for that group, you have to select game objects on your scene that have the TSTrafficLight component on it to be able to create the traffic light group.



The second menu option (Simple Vehicle) has one sub menu, Add Empty – This option adds an empty car to the scene and prompts a configuration window to add the car body, tires, etc. (This would be covered on how to Rig a car section).



The third menu option (iTS Manager) has three sub menu, Add iTS Manager to the scene, Add iTS Traffic Spawner to the scene and Add iTS Traffic volume to the scene, the first option would add an instance of the iTS Manager component into a new game object on the scene if there isn't any already. If there is already one, the following warning message would be displayed:



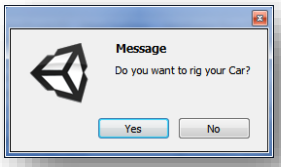
The second option would add the traffic spawner to the scener, this is needed to make the traffic system work.

And the third option adds a new traffic volume to the scene, if a game object is selected the traffic volume script would be added into a new game object that would become child of the selected object.

How to Rig a new Car

For rigging a new car you should use the Add Empty sub menu under the Vehicle menu option. This would display a window like the one on figure 1.

Note: When you press this option, the following Dialog box would appear, normally you should answer yes to this question, and a Rig car Wizard would be opened and would guide you through the Rig process. If you answer no, you need to do it all manually



If you press “Yes” the following window would appear:

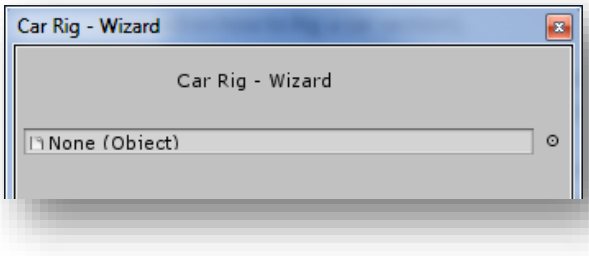
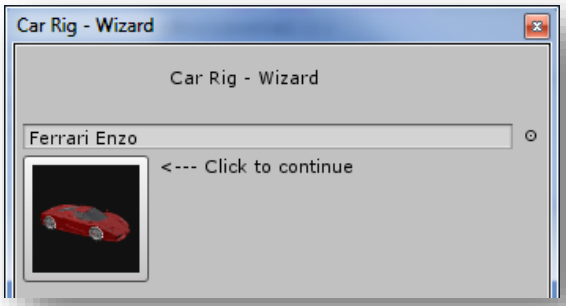


Figure 1

On this window you need to drag and drop the 3d Model you want to rig (car)

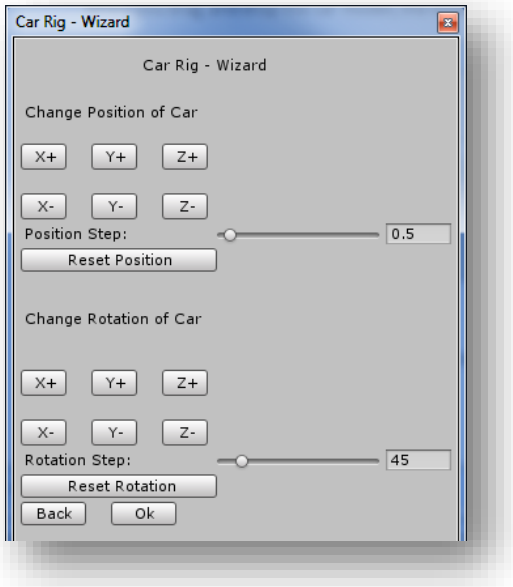
When you drag and drop the car model, the window would look like this:



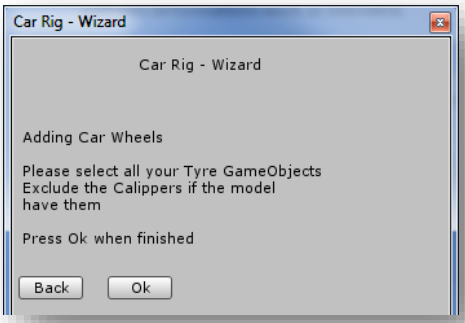
Now just click on the car preview button, if it is the model you want to rig, if it is not the model, Drag and drop the correct model to Rig.

The next step is to adjust the position and rotation of the model if is needed.

Note: Is better to do these adjustments on the 3D Modeler application (Blender, 3d Max, etc.) in order to set them correct before importing the 3d Model to Unity, this is to ensure that the collision deformations work as intended.

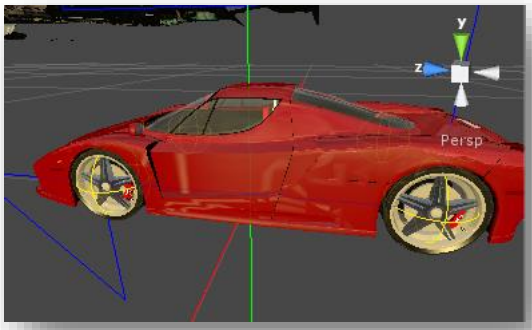


The next step is to assign the Tires, for this step you need to select all the game objects that conforms the tire and wheels and then press on the “OK” button like in the next picture:

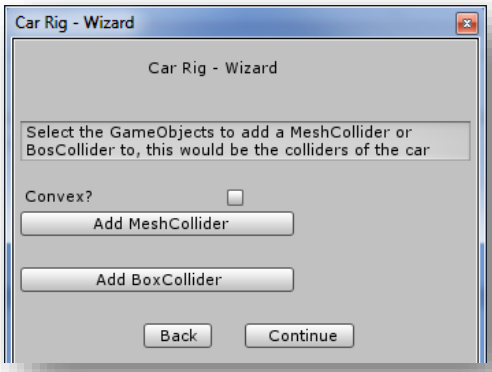


On the following window, you can set the wheel radius independently for the rear and front wheels, please consider that the Wizard already calculates the optimum wheel radius according to the wheel models selected previously.

Here is Picture of a car when you have selected the Tire Models, you can see that the Wizard marks the Tire radius in yellow spheres.

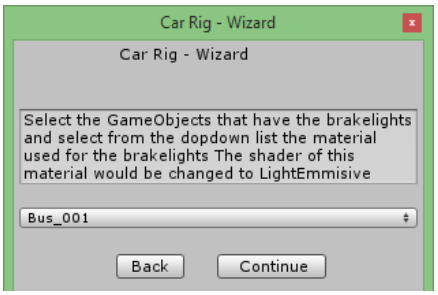


Adding the colliders for the model, to add colliders to the model, you can either add a Meshcollider or a BoxCollider, you just need to select all the objects of the car you want to add a collider to, and press on the desired button. Also you can select in the case of adding Meshcolliders, if you want it to be convex by selecting that option. When finished, press “Continue” Button.

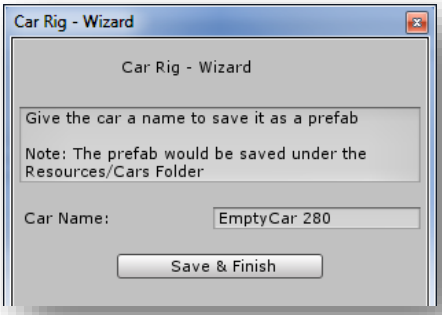


Adding Brake Lights, to add the brake lights just select all the gameobjects that represents the brake lights, then select the corresponding material from the dropdown list used for the brake lights in case the object has more than one material.

Then you can add the number of flares you want per object selected, and when finished press “Continue” button.

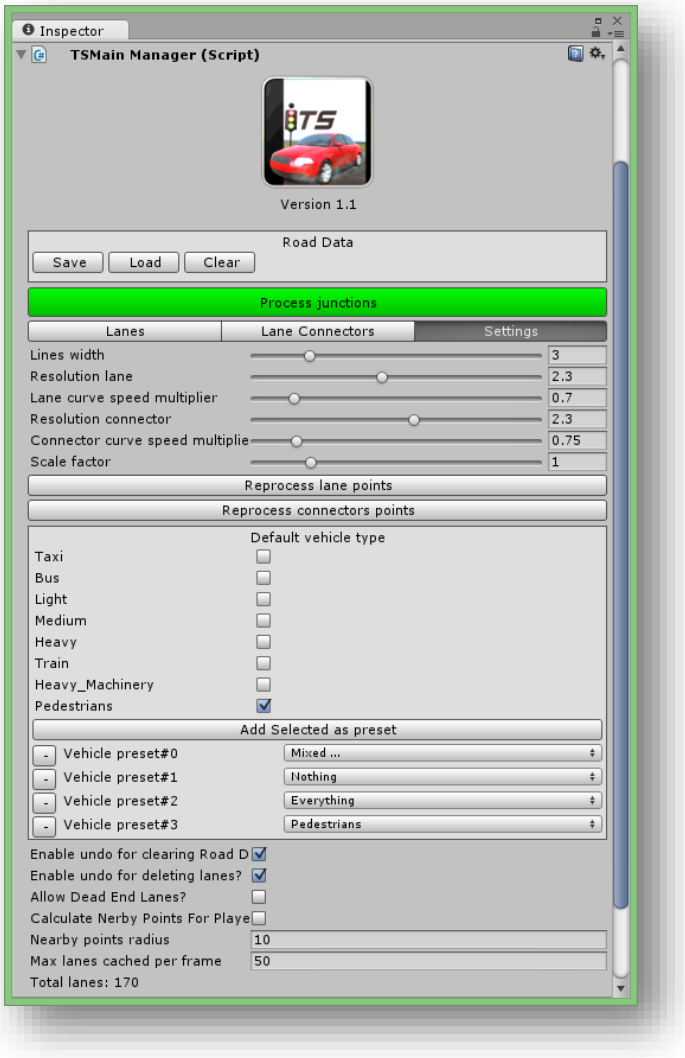


The last step, is just to click on the “Save & Finish” button, this would add a prefab of the model on the Resources/Cars folder on the project.



Traffic System Settings and Configuration

The Traffic System have some settings and configurations that can be changed to fit your needs, those settings can be accessed by selecting the iTSManager object on the scene and pressing the settings tab.



The first button “Process Junctions” needs to be pressed when the lanes are all placed and in order to use the traffic system at runtime.

The settings are the following:

Lines width: This option is for changing the visual width of the lanes while on edit mode.

Resolution lane: This options sets the distance between each graph point on each of the lanes, the bigger the value, the less points are used, and the smaller the more points are used.

Lane curve speed multiplier: This option is a multiplier for the max speed on the curves that the system calculates when the lanes and connectors are processed using the “process junctions” button.

Resolution connector: This options sets the distance between each graph point on each of the connectors, the bigger the value, the less points are used, and the smaller the more points are used.

Connector curve speed multiplier: This option is a multiplier for the max speed on the curves that the system calculates when the lanes and connectors are processed using the “process junctions” button.

Reprocess lane points: This button reprocess the lanes points, this must be pressed when the lane resolution is changed in order to apply the new resolution.

Reprocess connector points: This button reprocess the connectors points, this must be pressed when the connectors resolution is changed in order to apply the new resolution.

Default vehicle type: This toggles would be the default vehicles type that would be included when creating new lanes by default, change them to fit your needs.

Add selected as preset: This button would add the current vehicle types selected as a new preset that would be shown when a lane or connector is selected for a faster apply of the vehicle types that you want for your lanes and connectors.

Enable undo for clearing road data?: If this is enabled, the Unity undo would save all the info of the lanes the manager has before clearing the road map data, this could be a really slow and time consuming process if your scene has a lot of lanes.

Enable undo for deleting lanes?: If this is enabled, when you delete a lane you would be able to use Ctrl-Z to undo the deleted lane, this could take some time depending on the complexity and amount of lanes you currently have.

Allow dead end lanes? If this is enabled, the road map could have lanes that doesn’t have connectors, if enabled of disabled you must reprocess junctions.

Calculate nearby points for the player: If this is enabled (you must reprocess the junctions) the editor system would do a look up for each lane point to get the reference of all other lanes points that are within the Nearby points radius. This is just useful and used only for the TSPlayerFinder script, if you don’t or won’t use that script, keep this disabled to save memory space.

Max Lanes cached per frame: This is the amount of lanes that the editor would process on each frame to detect the lanes that are currently visible on the scene view.

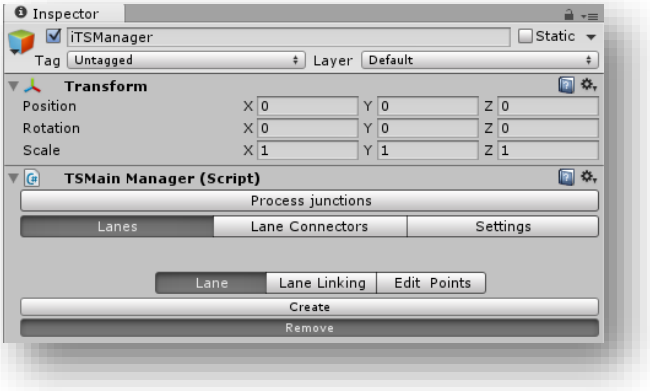
Lanes

Adding lanes

To add a new lane select the “Lanes” tab, with the “Lane” sub tab and the “Create” option as displayed bellow, then just with the mouse put the pointer where you want to start the lane and press the left mouse button and drag the mouse to the point where you want the lane to end and release the left mouse button.

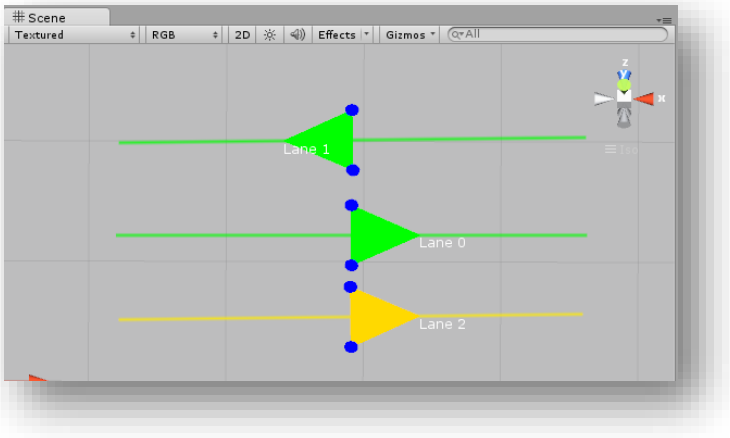
Removing lanes

To remove lanes, just select the “Lanes” tab, with the “Lane” sub tab and the “Remove” option and left click on the lane you want to remove.



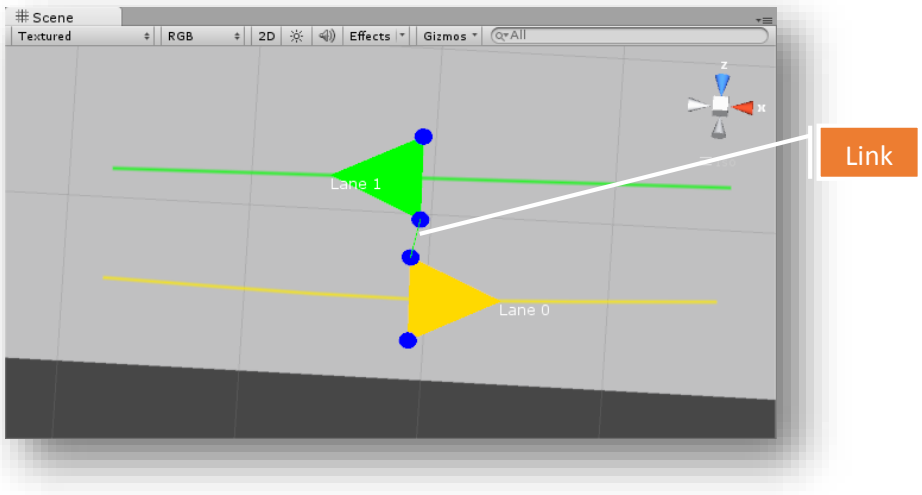
Linking lanes

You can link lanes together, this is useful to make possible overtaking and lane changing for the NPC traffic cars to be able to do so, you can link together the lanes by selecting the “Lanes” tab and the “Lane Linking” sub tab and the “Create” option, this would make 2 disc to appear on the lanes triangle that shows the lane directions, as on the following picture:

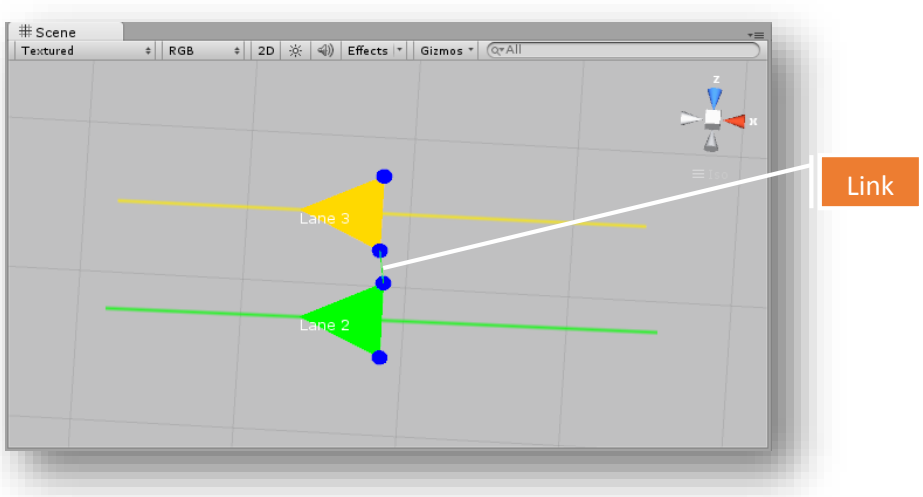


You can link together lanes that goes to the same direction (normal changing lane) and to contrary direction (overtaking lane), to add a new link, just make sure to have the options mentioned before selected and the “Create” option and simply click on the side of the lane you want to link with a left mouse button and hold and drag to the other lanes point as shown on the image bellow:

Overtaking linking (this is for the cars change to a contrary lane to overtake a car)

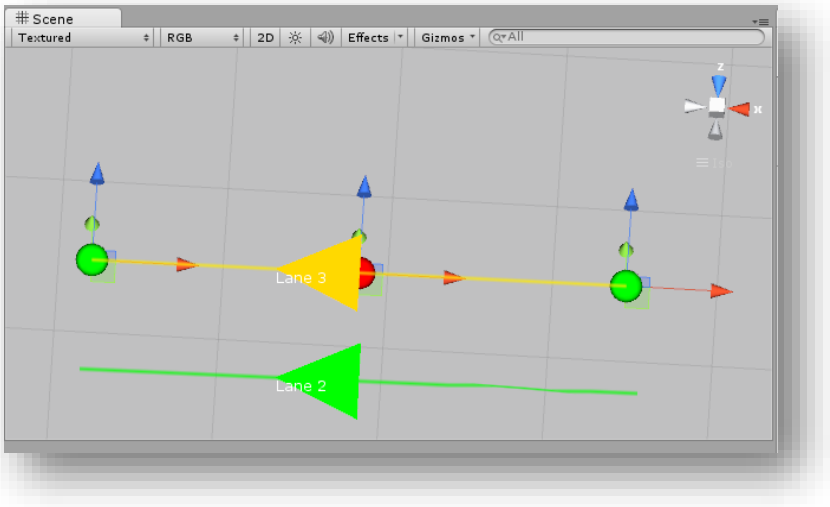


Normal Lane change linking (this is for also overtaking another car but the lanes are on the same direction)



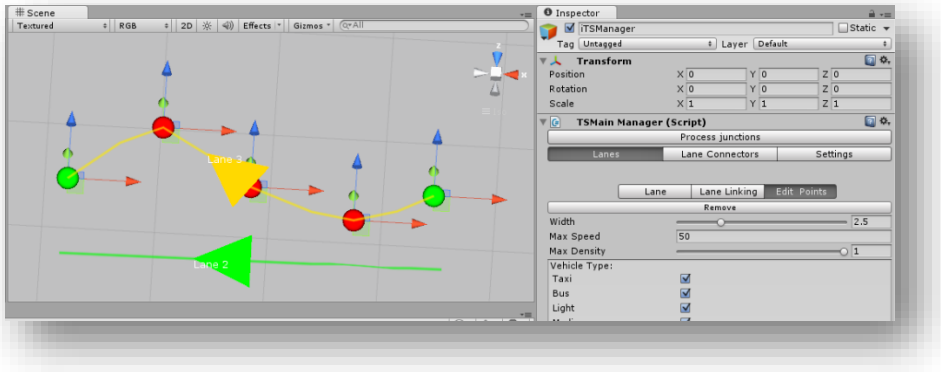
Editing lane shape (editing the lane points)

You can always change the shape of the lanes that are already created, this is done by selecting the “Lanes” tab and selecting the lane you want to edit by pressing the left mouse button on top of the lane you want to select (the lane would change its color to blue, when the mouse is over the lane and to yellow if the lane has been selected), then select the “Edit Points” sub tab, this would show the following handles on the scene view, which you can drag to move the lane points and change the lane shape.



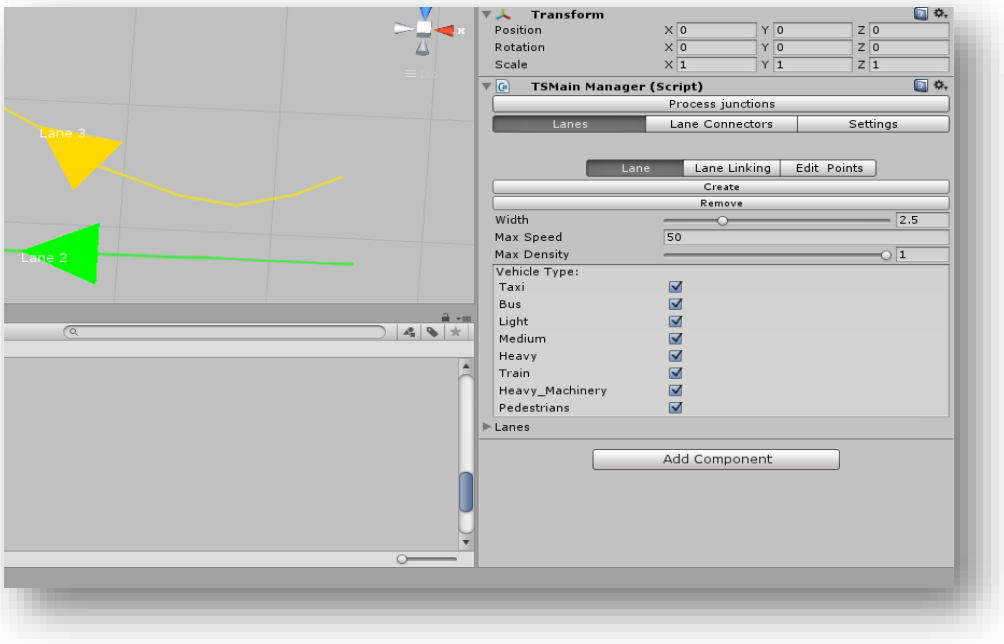
Adding/Removing lane points

You can also add or delete the lane points to change its shape, this is done by selecting the “Lanes” tab and selecting the lane you want to edit by pressing the left mouse button on top of the lane you want to select (the lane would change its color to blue, when the mouse is over the lane and to yellow if the lane has been selected), and the “Edit Points” sub tab, then if you want to add a new point, just do a left click mouse on the scene vew where you want to put the new point, if you want to remove a point, just select the “Remove” option and left mouse click on the point you want to delete (only the red points can be deleted from a lane).



Lane settings

There are some settings that can be tweaked on each lane, the settings are shown on the following picture:



You can access the lane settings by selecting the lane you want to change the settings, having the "Lanes" tab selected and the "Lane" sub tab selected and then left mouse clicking the lane you want to select, the available settings are:

Width: This changes the lane width.

Max Speed: The max speed for this lane.

Max Density: The % of this lane ocupation to spawn cars on it, if it is set to 1 the 100% of this lane could be ocupied when spawning cars on it, if less (i.e. 0.5) the 50% of occupation would be use to spawn cars on this lane.

Vehicle Type: The types of vehicles that can drive over this lane.

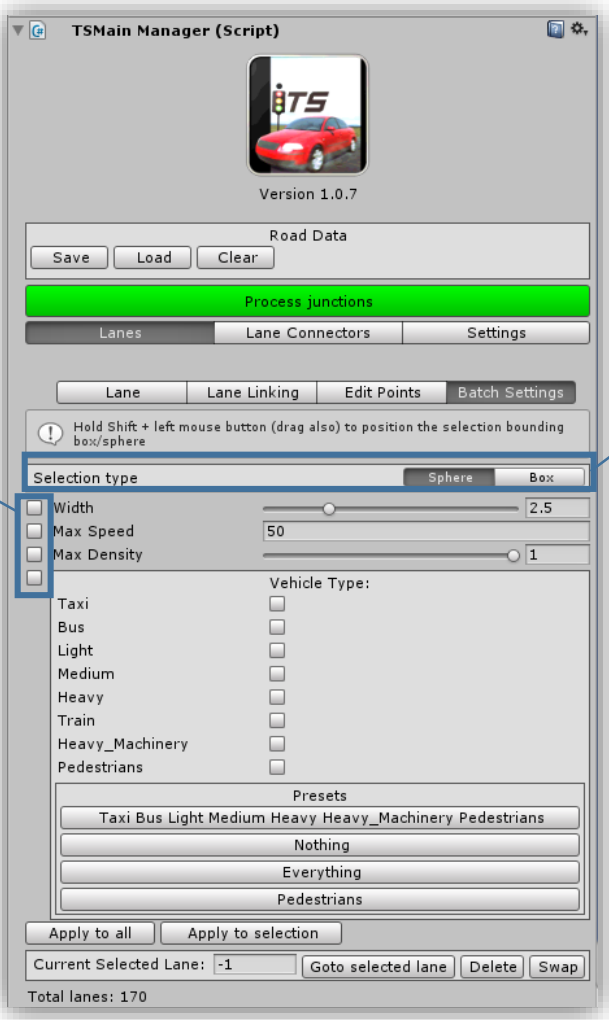
Presets: The vehicle type presets available to set to this lane, click on the desired preset to apply it.

Batch Lane Settings

You can batch settings by selecting the "Batch Settings" option from the Lanes toolbar, then you can apply those settings to all lanes or to the selected lanes.

You can select lanes using the bounding box or the sphere.

With these toggles you can select the settings to apply to the lanes



Select the selection shape, either a bounding box or a sphere

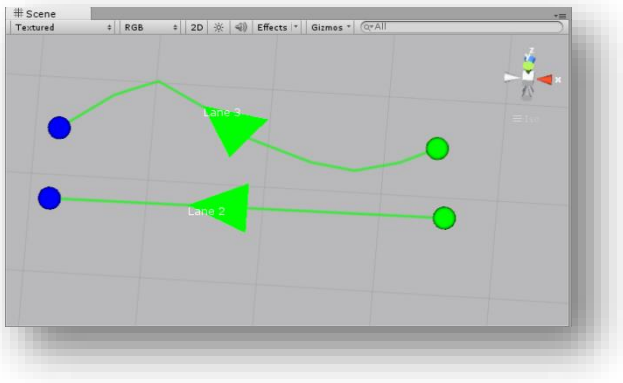
Connectors

Adding connectors

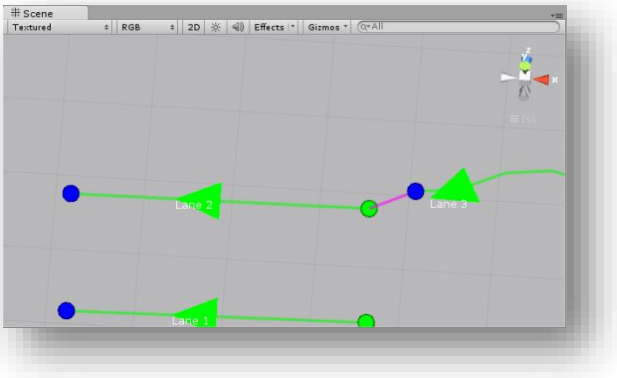
The connectors can be added in order to join two lanes together, to make possible for the traffic cars to go from one lane to another.

To add a new connector you need to select the “Lane Connectors” tab and the “Connector” sub tab, then select the “Create” option.

This would make the lanes on the scene to have two spheres at the start and ending points one green (starting point of the lane) and the other blue (see picture below).

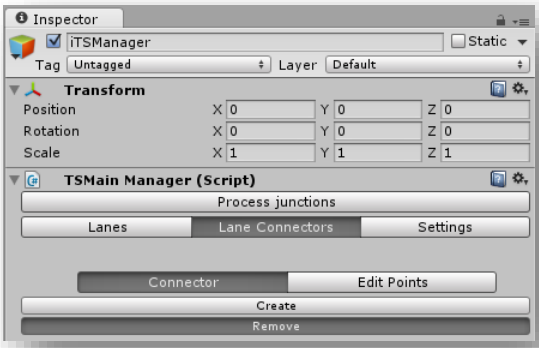


You can place a connector between two lanes only from the blue to the green spheres, to do this just click on the first sphere and keep the left mouse button down and drag the mouse to the desired point of the other lane and release the left mouse button.



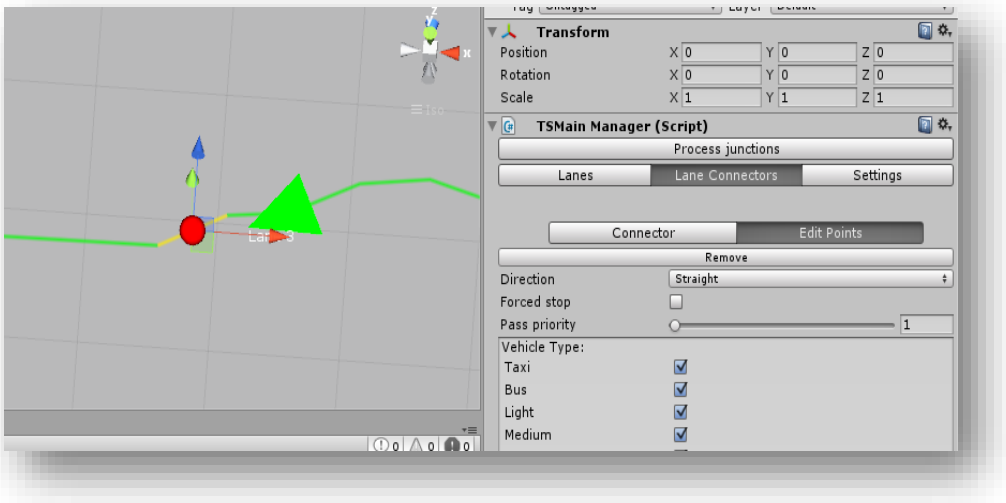
Removing connectors

For removing the connectors, you need to select the “Lane Connectors” tab and the “Connector” sub tab, then select the option “Remove” and do a left mouse button click on the connector you want to remove, the connector would be blue if the mouse is on it, otherwise it would be magenta, click on the connector you want to remove if it is blue.



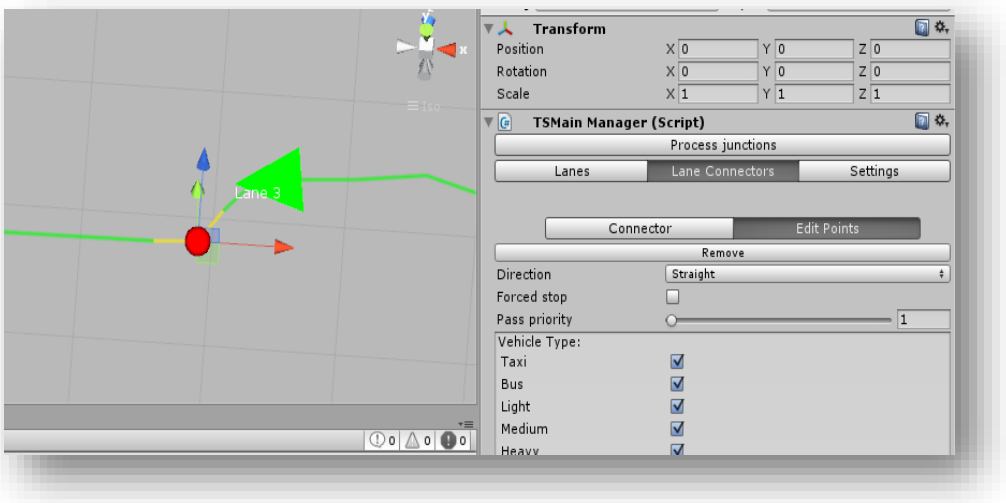
Editing connector shape(editing the connector points)

It is possible to change the shape of the connector at any time on the Unity editor, to do so, select the “Lane Connectors” tab and the “Connector” sub tab, then left mouse click on the desired connector you want to edit and select the “Edit Points” sub tab, this would show all the editable points of the current selected connector, and you can move them to change the shape of the connector to fit your needs.



Adding/Removing connector points

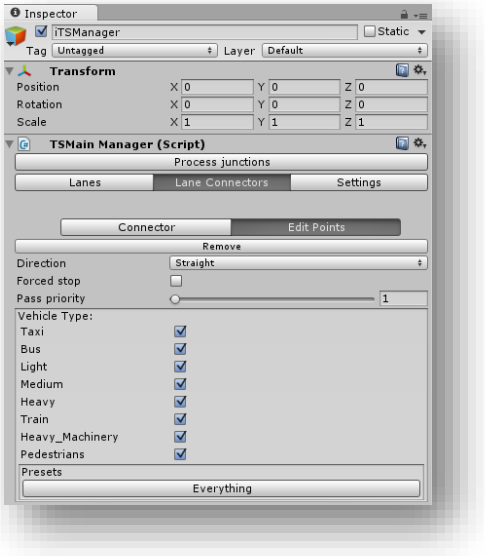
You can also add or delete the connector points to change its shape, this is done by selecting the "Lane Connectors" tab and selecting the connector you want to edit by pressing the left mouse button on top of the connector you want to select (the connector would change its color to blue, when the mouse is over the connector and to yellow if the connector has been selected), and the "Edit Points" sub tab, then if you want to add a new point, just do a left click mouse on the scene vew where you want to put the new point, if you want to remove a point, just select the "Remove" option and left mouse click on the point you want to delete (only the red points can be deleted from a connector).



Connector settings

There are some settings that can be tweaked on each connector, the settings are shown on the following picture:

There are some settings that can be tweaked on each lane, the settings are shown on the following picture:



You can access the connector settings by selecting the connector you want to change the settings, having the “Lane Connector” tab selected and the “Connector” sub tab selected and then left mouse clicking the connector you want to select, the available settings are:

Direction: This is the relative direction from the lane this connector starts and to where the next lane is, letting know the traffic cars if the next lane is straight ahead, to the left or to the right, this is needed to make possible the use of turning lights for the traffic cars.

Forced stop: If this option is enabled, the cars would make a stop before trying to cross this connector, this is to simulate Stop signs.

Pass priority: This is a proirity of this connector to let know the traffic cars if they should yield for another connector with higher priority than this one.

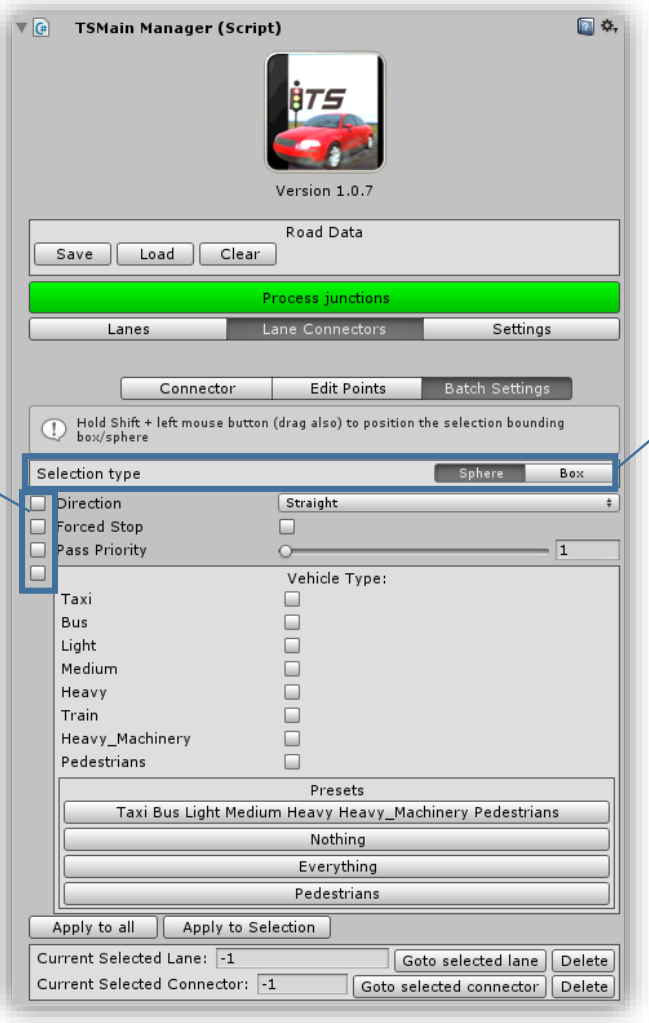
Vehicle Type: The types of vehicles that can drive over this connector.

Presets: The vehicle type presets available to set to this connector, click on the desired preset to apply it.

Batch Connector Settings

You can batch settings by selecting the “Batch Settings” option from the connectors toolbar, then you can apply those settings to all connectors or to the selected connectors.

You can select connectors using the bounding box or the sphere.



With these toggles you can select the settings to apply to the connectors

Select the selection shape, either a bounding box or a sphere

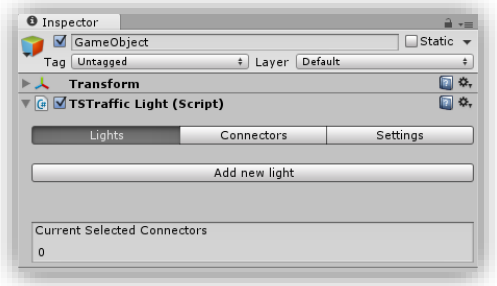
Traffic lights

The traffic lights can be added to game objects in order to control the traffic flow on some selected points of a lane or connectors, this is useful to control the traffic on junctions and some other points on the traffic system.

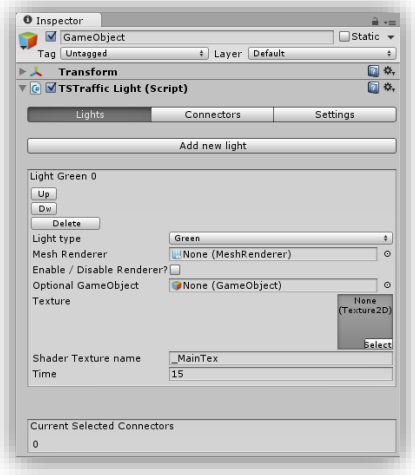
The traffic light script can be added by using the unity built in add component option on the selected game object or by using the provided option on the unity menu GameObject->iTS->Traffic lights->Add traffic light script to selected.

Adding Traffic light lights (Red, yellow, green, etc)

To add a new light to an existing traffic light, you need to select the traffic light on the scene view and then on the inspector click the button “Add new light”, this would create a new light and would be shown its options on the inspector as on the picture bellow:



No lights added

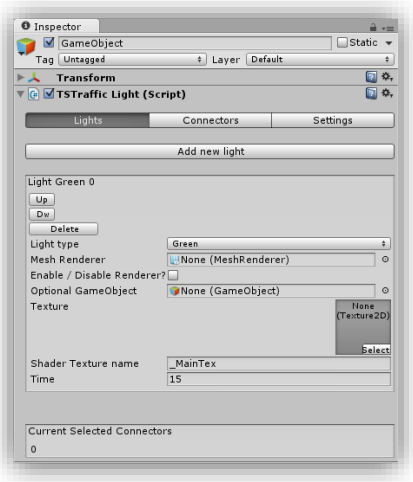


One light added

Light fields and options

The traffic lights can have as many lights as needed, each light have their own options, this allows to create different types of traffic lights in a very custom way.

The options available for each light are the following:



Up: This button would move up the light on the list.

Dw: This button would move down the light on the list.

Delete: This button would delete the light from the list.

Light type: The type of the light, this include red, yellow, green, turning red, turning yellow, turning green and no lights.

Mesh Renderer: This is the renderer this light would control.

Enable/Disable Renderer?: This option if enabled, would make the renderer on the Mesh Renderer field be enabled or disabled when this light is enabled or disabled correspondingly.

Optional GameObject: This field is for adding here a game object that you want to be enabled and disabled when this light does the same.

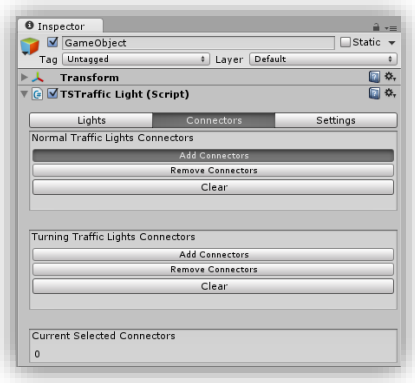
Texture: The texture that would be assigned to the Renderer when this light is on, this is not used if the Enable/Disable Renderer option is used.

Shader Texture name: The name of the texture shader property, so this light could swap the texture when it's on.

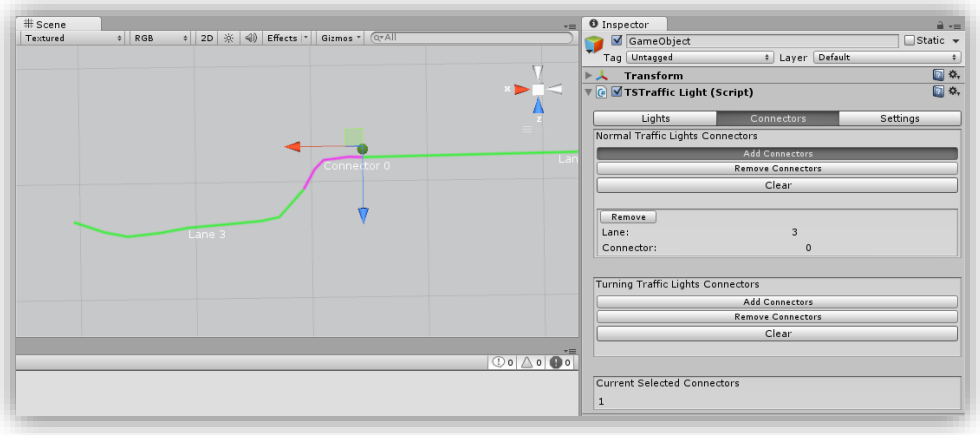
Time: The time this light would be on (in seconds).

Selecting Traffic light connectors

In order for the traffic light to actually control a connectors, you have to assign to it the points you want it to control, for this, you have to select the “Connectors” tab on the inspector and select the “Add Connectors” option, then just left mouse click on the connector you want to add to this traffic light.



Add connector option selected



The selected connector added

You can notice that when a connector is added, its color changes to magenta, and a new item appears below the “Add connectors” option with a button to remove the just added connector.

Traffic light general settings

The traffic lights have some general settings, to change these settings select the “Settings” tab on the inspector, this would show you 2 options, as on the following picture:

Options description:

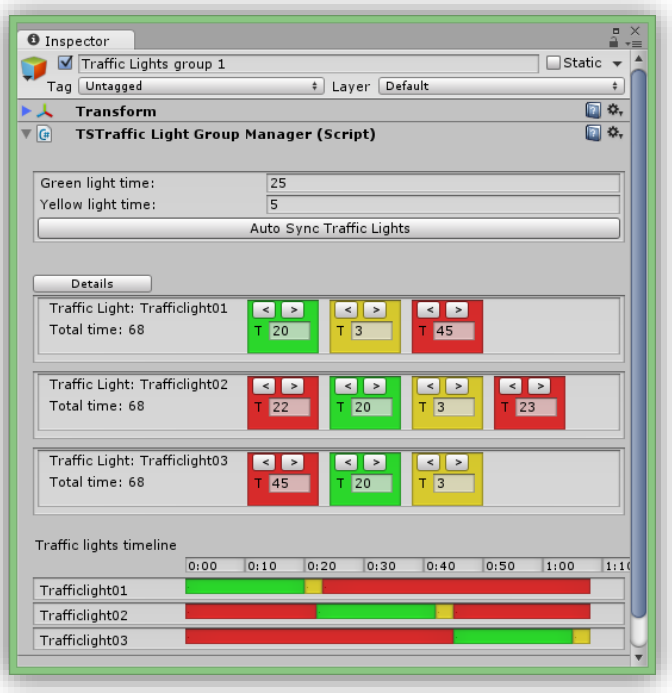
Yellow Lights Stop Traffic?: If this option is enabled, the traffic cars would stop if the yellow light is on, otherwise the cars would only stop if the red light is on.

Light Range: This is a distance in meters that is used to show the surrounding lanes and connectors within that distance from the traffic light at editor mode, this is to avoid the scene becoming too heavy, and just showing the closer lanes and connectors to the selected traffic light.

Traffic light groups

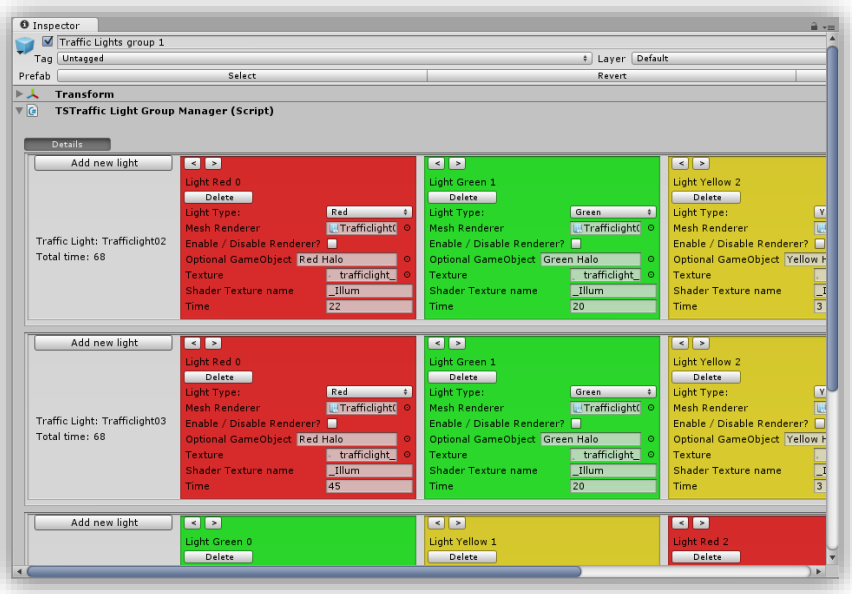
The traffic light group is a way to group traffic lights to make their setup (as a group of traffic that would have to be synced) easier and more visual, to group traffic lights you just need to select all the gameobjects that have the TSTrafficLight component on them and that you want to sync and then select the menu option GameObject→iTS→Traffic Lights→Group and this would create a new game object and would make all the selected traffic lights its children.

The inspector of the traffic light group gives the ability to also tweak individual traffic lights settings right on the inspector and also visually see the timeline of all the traffic lights to be able to sync them easily, here is a pic of a traffic lights group:



With the “Auto Sync Traffic Lights” button you can sync the traffic lights in a single click, for now they would be ordered one after the other using the times provided on the fields before the button (traffic lights gets sync in a cascade kind of way). More sync options would be added on future releases.

As you can see on the screenshot, you can change the individual lights times of each traffic light so they match and sync, and visually see the changes on the timeline, also if you click on the “Details” button, the small colored square areas of each light would expand and show all the individual settings of each of them, as on the following picture:



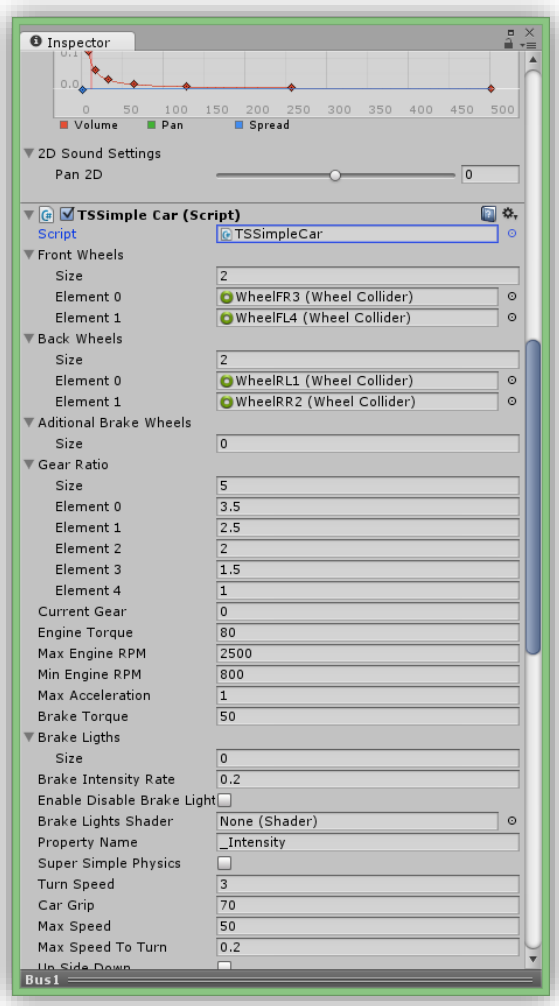
This gives the advantage of been able to tweak any of the traffic lights settings without having to leave the group, or having to select the individual traffic light to change its settings.

Traffic cars physics settings

TSSimpleCar component settings

The traffic system comes with a simple car physics that uses wheelcolliders and also have an even simpler physics that can be made to just use a collider and a Rigidbody (suitable for mobile platforms).

The settings available for this physics is pretty simple to setup, and the code is open source for it, so you can modify it to fit your needs.



The available settings are:

Front wheels: This array should contain all the front steered wheels of the vehicle.

Back wheels: This array should contain all the back non-steered wheels of the vehicle.

Additional brake wheels: This array is for putting here any additional wheels that would also bake the car, i.e. the wheels of a trailer for a truck.

Gear Ratio: This array is for setting the gear ratios of this car.

Current Gear: This is the actual gear the car is on.

Engine Torque: The total torque of the car.

Max Engine RPM: The maximum engine RPM.

Min Engine RPM: The minimum engine RPM.

Max acceleration: This value determine how fast the car can accelerate, the higher the value the faster the car accelerates.

Brake torque: The maximum torque for the brakes.

Brake Lights: This array can be filled with the brake lights renderers to make them turn on and off.

Brake intensity rate: This value controls how fast the brake changes when they are pressed of depressed.

Enable Disable Brake Light: This option would make the brake light renderers to just be switched on and off when the brake lights are on or off.

Brake Lights Shader: This is for assigning here the shader name if the Enable Disable Brake light option is not used, and instead use a shader property to manage the brake light.

Property name: The name of the shader property that would be used to turn on and of the brake lights.

Super Simple Physics: This enables the super simple physics, which does not use wheel colliders.

Turn speed: The max turning speed of this car, for the super simple physics mode.

Car Grip: The car side grip, for super simple physics mode.

Max Speed: The max speed of this car, for the super simple physics mode.

Max Speed to turn: The maximum speed to start turning the car in super simple physics mode.

Upside Down: This is a bool variable to inform if this car is upside down.

Crashed Smokes: This array holds one or more particle systems to simulate smoke from the engine when this car crashes.

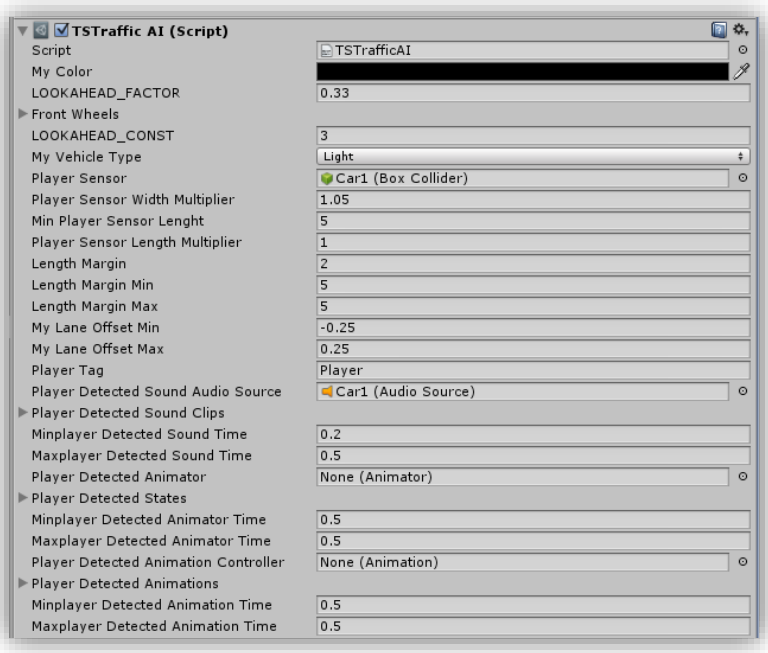
Turn right light: This is the GameObject that holds all the right turning lights of this car, this game object and all its children would be activated and deactivated to simulate the turning light.

Turn left light: This is the GameObject that holds all the left turning lights of this car, this game object and all its children would be activated and deactivated to simulate the turning light.

CoM: The center of mass position, this is taken from a gameObject that needs to be child of the car main game object.

Traffic Cars traffic scripts settings

The traffic cars have some few parameters that can be setup to get some particularity on the traffic simulation, those parameters are as on the following picture:



On the class TSNavigation there is no parameter to setup.

TSTrafficAI class

On this class we have the following parameters:

LOOKAHEAD_FACTOR: This is the factor (based on the speed of the car) that the car would be pointing at the current waypoint, the greater this value the more farther would be the current waypoint in front of the car at higher speeds.

Front Wheels: On this array should be assigned both of the front wheels transform, it is used to calculate the center between the front wheels.

LOOKAHEAD_CONST: The default distance the current waypoint is from the car.

My Vehicle Type: The type of this vehicle, should be Racers.

Player Sensor: This is the box collider (trigger must be enabled) that would be used to sense were the player is.

Player Sensor Width Multiplier: This value would be multiplied by the calculated car width and applied to the player sensor collider size on the x axis.

Min Player Sensor Length: This is the minimum player sensor length when the car is at low speeds or stopped.

Length Margin: This is the min distance between the traffic cars(not used, now it is set with the length margin min and max.

Length MarginMin: This is the min distance between the traffic cars.

Length MarginMax: This is the max distance between the traffic cars.

My Lane Offset Min: The minimum offset that this vehicle could have, this is good to give some variation to the Vehicles position on the lane, so they don't look like they all follow exactly the same line.

My lane offset Max. The maximum offset that this vehicle could have, this is good to give some variation to the Vehicles position on the lane, so they don't look like they all follow exactly the same line.

NOTE: if both MyLaneOffsetMin and max are set to 0 this feature is disabled.

Player Tag: The tag of the player car, this is to be sure the traffic cars detects correctly only the player car.

Player Detected Sound Audio Source: This would be the audio source that the car would use to play the audio clips when it detects the player car heading towards this car.

Player Detected Sound Clips: This is an array of audio clips that would be played randomly if the condition described on the option before is meet.

Min/Max Player Detected Sound Time: This would be the range that would be played the audio clips between each play (this is the time between each play of a random audio clip, the time is a random value between this min and max values).

Player Detected Animator: This is the Animator component that would be used to play the animations when the player is detected and is heading towards this car.

Player Detected States: The Mecanim states names that would be played randomly in case the previous case is meet.

Min/Max Player Detected Animator Time: This would be the range that would be played the states between each play (this is the time between each play of a random state, the time is a random value between this min and max values).

Player Detected Animation Controller: This is the Animation Component (Legacy Animation System) that would be used to play the animations when the player is detected and is heading towards this car.

Player Detected Animations: The Animation clips names that would be played randomly in case the previous case is meet.

Min/Max Player Detected Animation Time: This would be the range that would be played the animation clips between each play (this is the time between each play of a random animation clip, the time is a random value between this min and max values).

Spawner Settings

Cars (Array): this is the array of the cars you want the spawner to spawn on the scene, these cars needs to be rigged first with the car rigger wizard.

Amount: This is the amount of cars you want on the scene

Max Distance: This is the max distance of the spawning area, this match the outer green wire sphere you see on the scene view, on the position of the spawner.

Offset: This is the min distance offset of the spawning area, this match the inner green wire sphere you see on the scene view on the position of the spawner, the space between the outer and inner spheres is the area where the traffic cars would get spawned.

Closer Range: This is a range used for prototyping a feature where the cars gets super simple physics if they are farther than this distance from the spawner, this is with the objective of saving

performance with farther cars using a less CPU consuming physics, usually this is not needed on PC builds.

Refresh Time: This is the refresh time for the spawner to look for processing the points that are inside of the spawner area.

Manager: The reference of the iTSManager this spawner is locked to (this is usually got automatically)

Unused Cars Position: This is the vector3 position in global space, where the cars are put when they are not used on the scene (they are on the pool for been respawned later).

Respawn if upside down: If this is enabled and for any reason a traffic car gets upside down, they would get respawned.

Respawn upside-down time: How much time the car needs to be upside down to get respawned.

Respawn Altitude: The respawn altitude of the cars.

Disable Multi-threading: If this is enabled, the process of getting the actual points that are on the spawning area would be done on the main thread.

Global Point Offset: This is a global offset that is useful if your scene is too big and you start having floating point issues on the physics, and you may need to move the world to the center respect to the player Point of view.

Cars checked per frame: The amount of cars the Spawner would check if they are outside the spawning area to disable them per each update cycle.

Initial not spawning radius: This is the radius the spawner would check from its transform position, to avoid cars getting spawned inside this area, useful to avoid initial cars from getting spawned on top of the player.

Transform spawning check: This is a list of transform that would be checked by the spawner when is trying to spawn a car, and would avoid to spawn the car inside the defined radius of these transforms.