Vehicle Editor Resource Pack

User Guide

Created By: A-Lab Software Limited

Date Created: May 2010

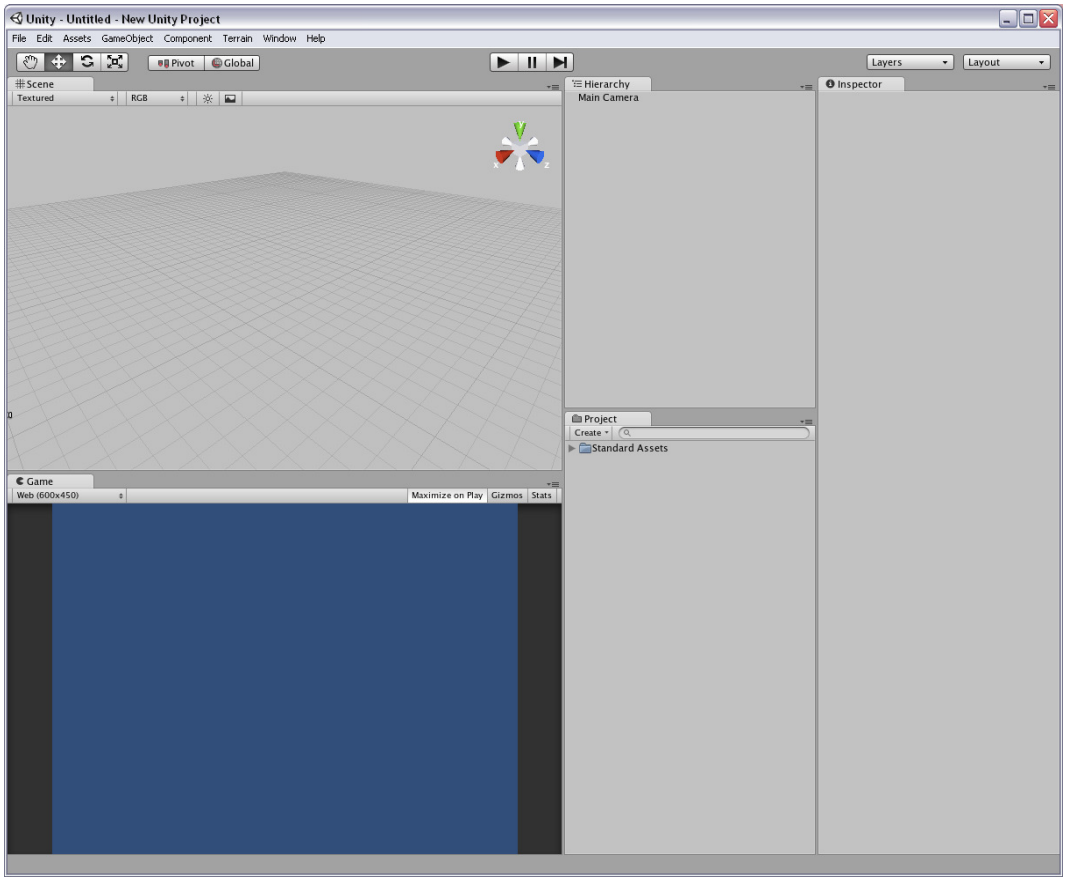
Date Modified: July 2010

Revision: 1.5

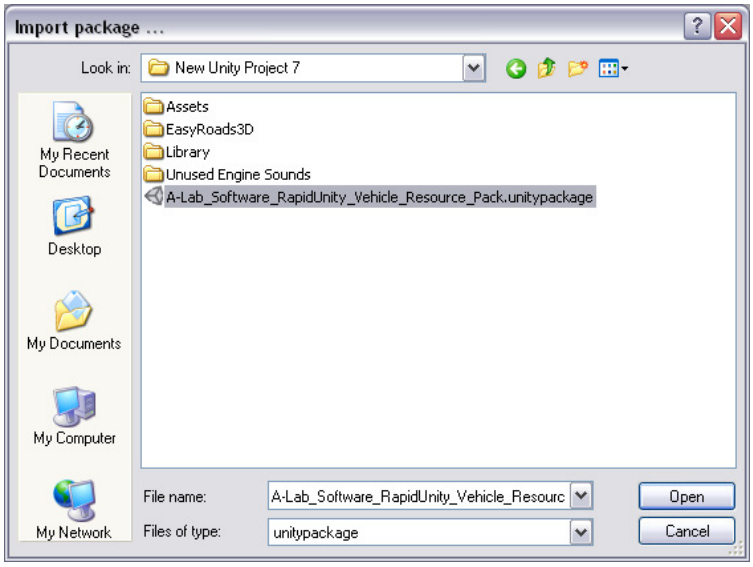
Table of Contents

Installation

1. Create a New or Open an existing Unity Project within the Unity Editor.

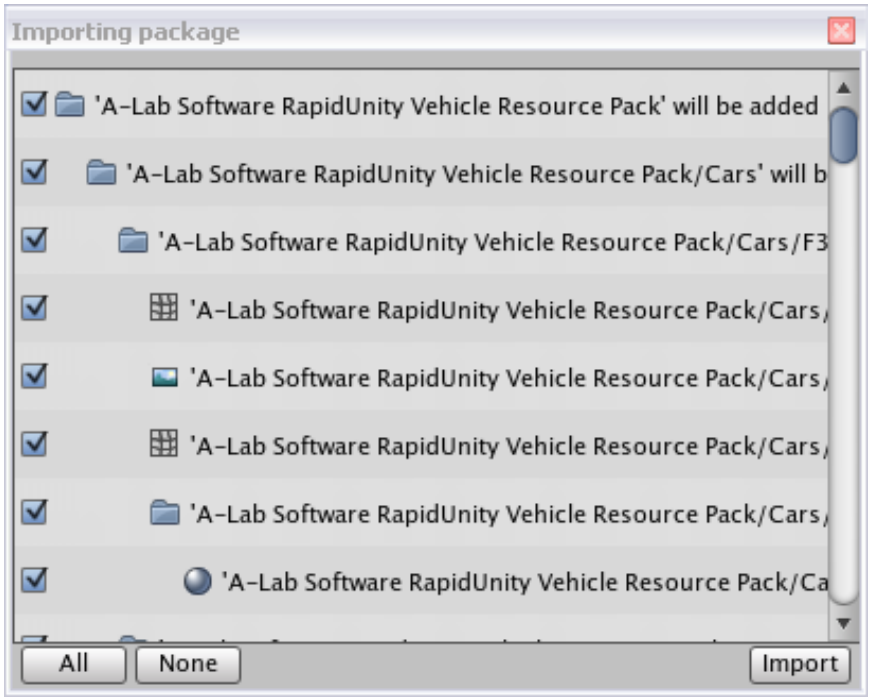


2. Next import the RapidUnity Vehicle Editor Resource Pack, by clicking the Assets menu, and selecting Import Package…

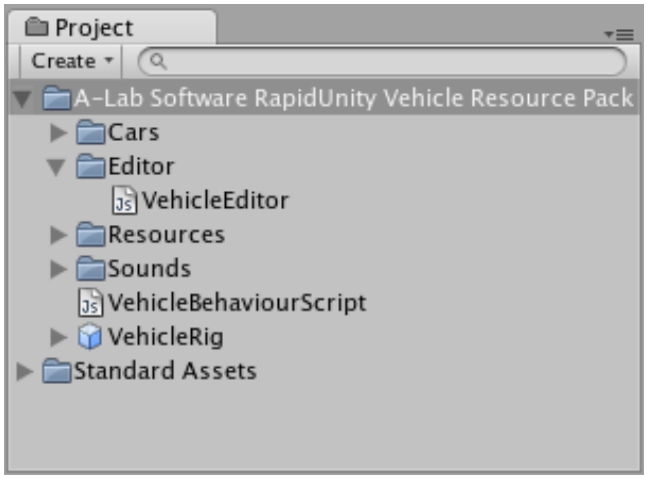


Select the A-Lab_Software_RapidUnity_Vehicle_Resource_Pack.unitypackage file and click Open.

3. Now the Importing package dialog will appear, simply click the Import button.



4. Once the import completes, your Project panel should contain the A-Lab Software RapidUnity Vehicle Resource Pack, and sub-folders.



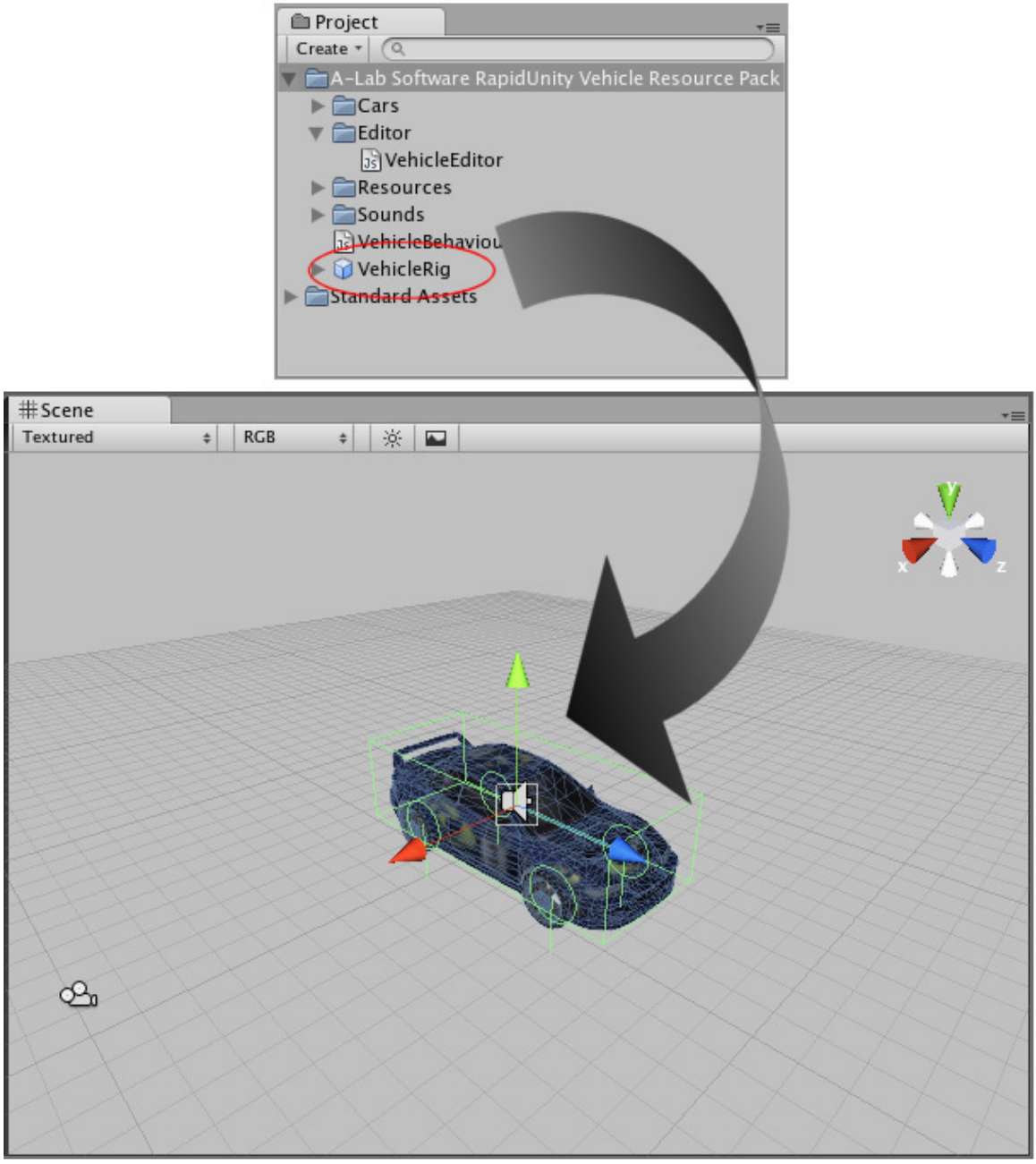
5. That's it! Installation is complete, now continue to the next section to learn how to use the resource pack.

How to use the Rapid Unity Vehicle Resource Pack in Unity

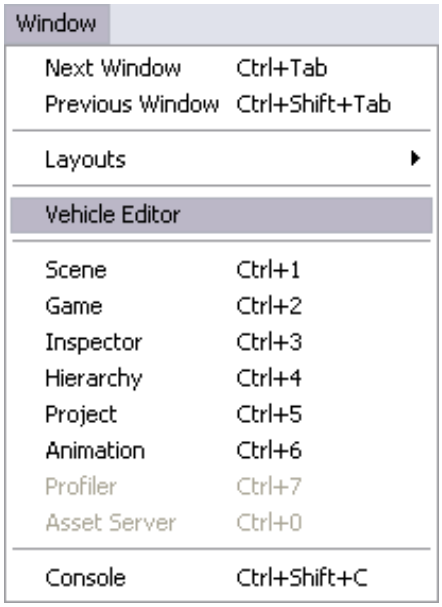
Using the RapidUnity Vehicle Resource Pack is very simple, by following the steps below you will be test driving your latest vehicles in no time!

Or if you prefer, you can simply open either of the two included Unity scenes named TestTrack1 and TestTrack2 found in the Scenes folder in the Project panel, and start driving the pre-configured vehicles around the scenes immediately!

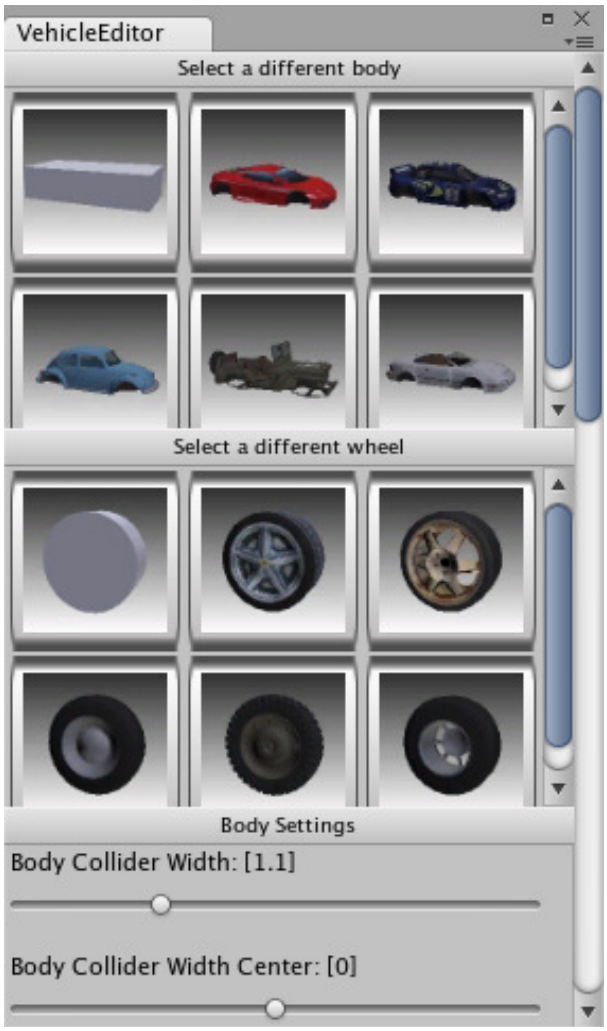
1. The first thing you need to do is add the Vehicle Rig to your scene, simply drag the VehicleRig prefab from the Project panel into your scene.



2. Next, you need to open the Vehicle Editor, to do this simply click the Window menu, and select Vehicle Editor.

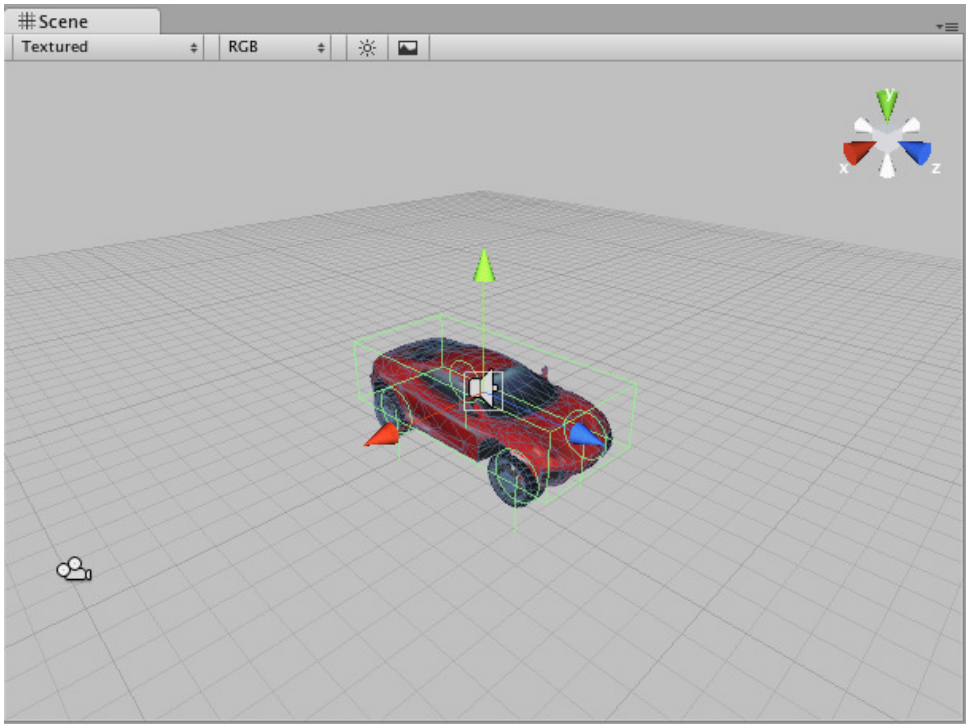


3. Now the Vehicle Editor panel will be displayed.



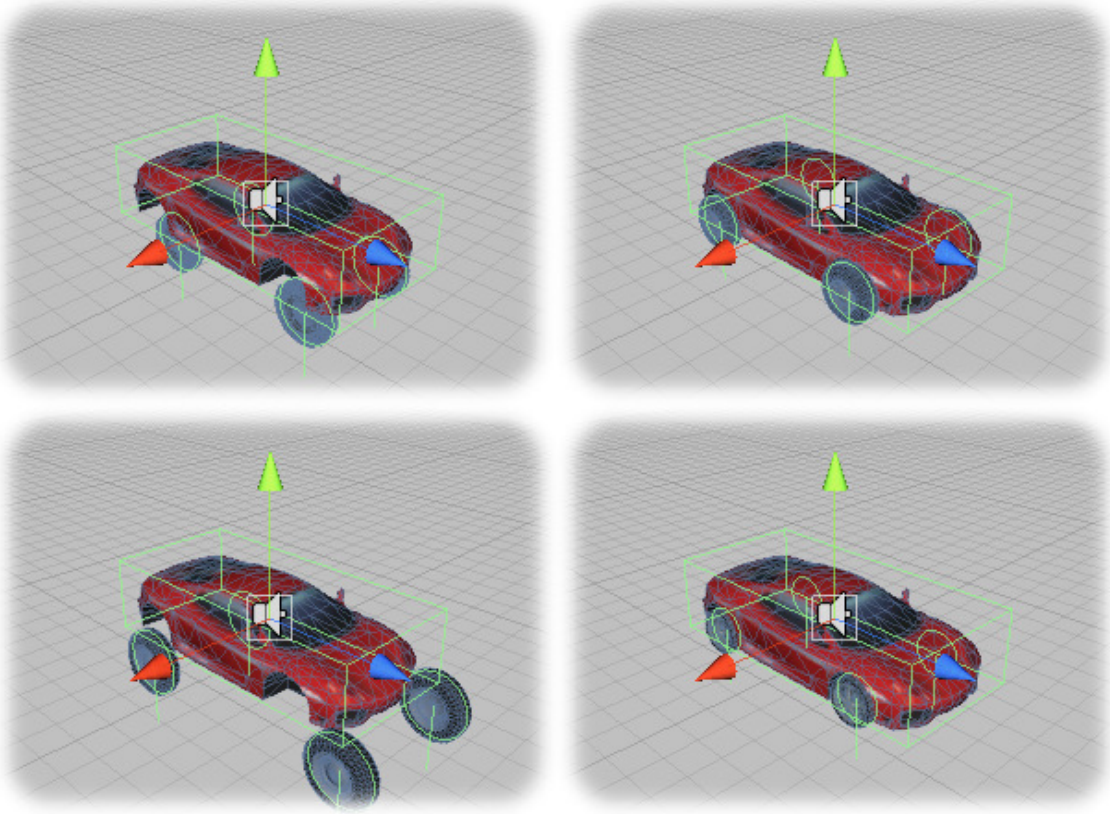
4. The Vehicle Editor panel is very easy to use, IMPORTANT make sure that the VehicleRig is selected in the Hierarchy panel.

5. Now click on one of the different vehicle body buttons in the Vehicle Editor panel, try the Red Ferrari for example.



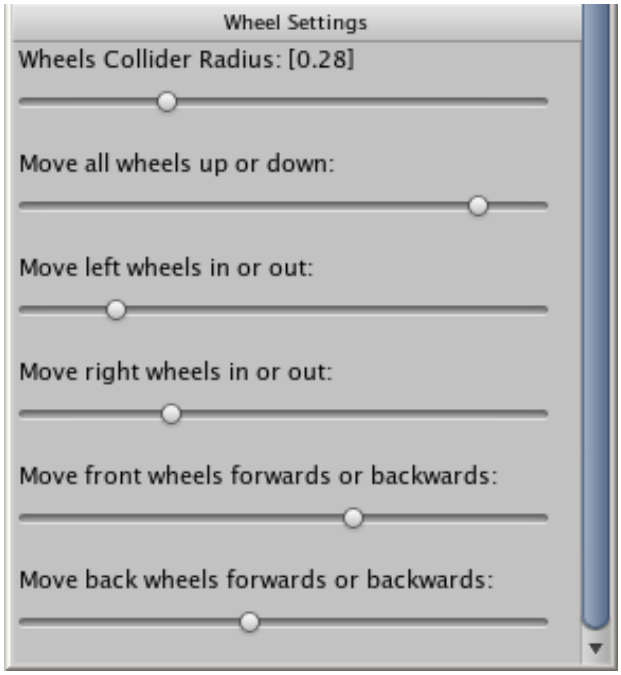
As you can see, the body of the VehicleRig has now been replaced with the Red Ferrari, notice also that the green box collider has also changed size and position, to fit the newly selected vehicle body.

6. Now try out some different wheels on your vehicle, simply click on each wheel button in turn to see the different types of wheels available.



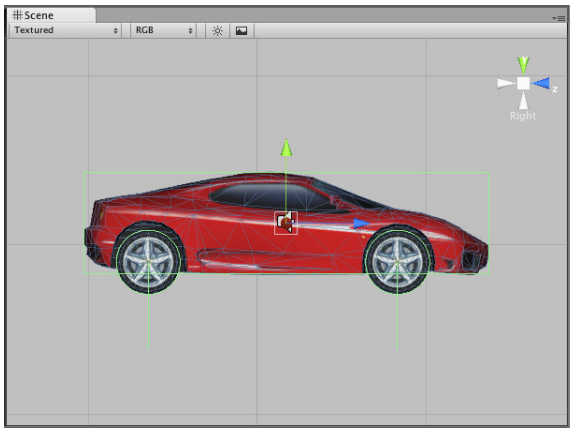
You will notice that the green wheel colliders automatically change size and position to match the newly selected wheels, do not worry that some of the selected wheels are not quite in the correct position, the Vehicle Editor makes it very easy to move the wheels into the correct position.

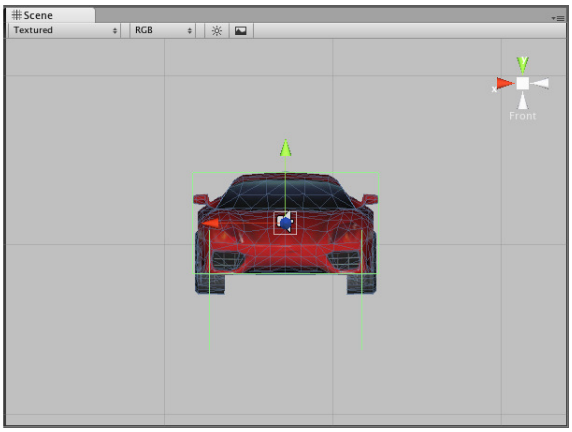
7. To change the position of the wheels, simply use the Wheel Settings section of the Vehicle Editor found at the bottom of the panel, use the scroll bar on the right-hand side to scroll down to the bottom.



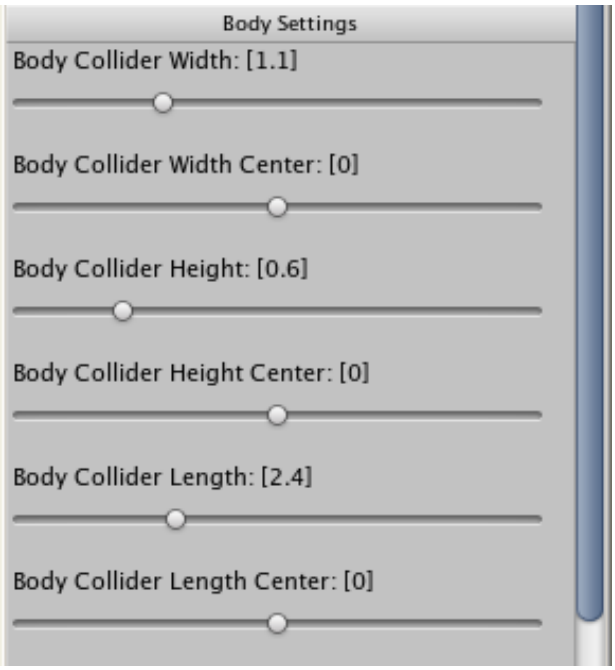
Have a play with the different slider bars to see how they affect the wheels.

TIP - Try changing the Scene view in Unity to side and front, it really helps when setting up a new vehicle.

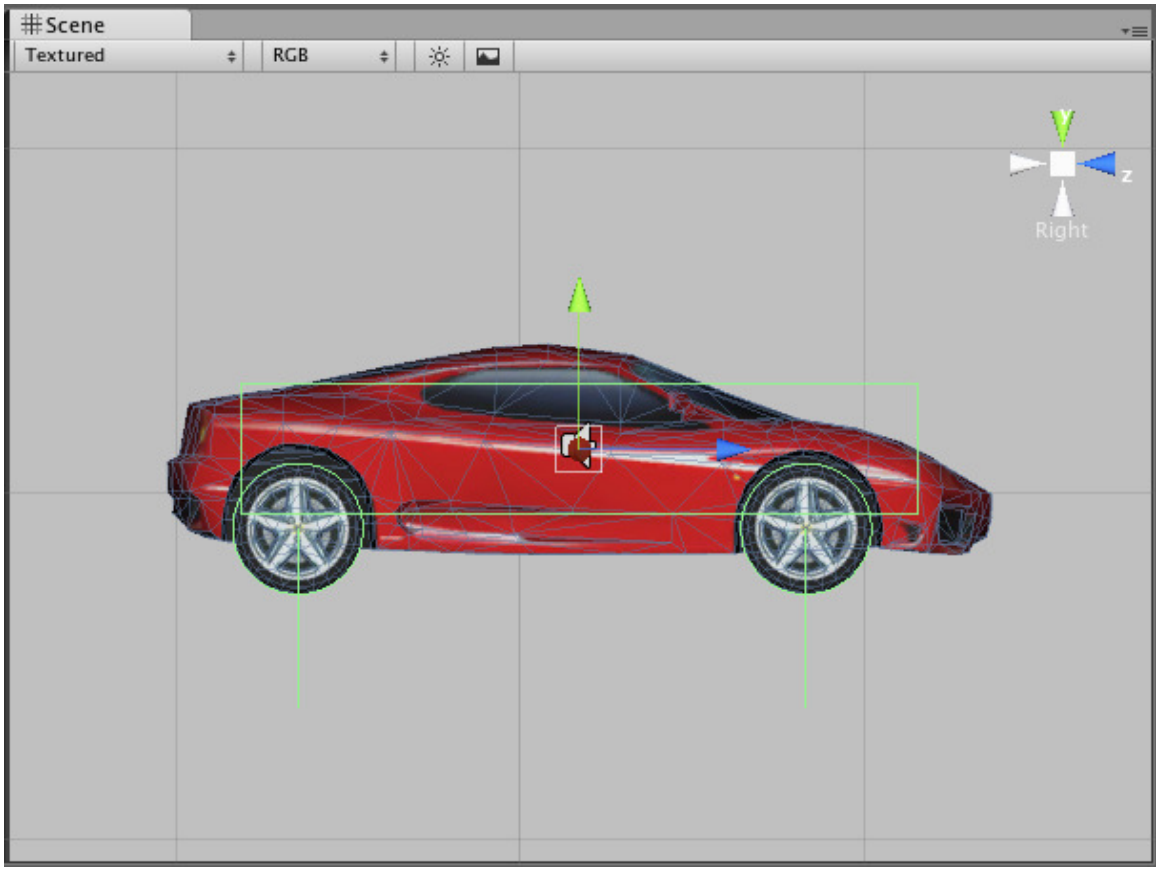




8. There are two other sections within the Vehicle Editor panel that we have not covered yet, the first is Body Settings, this section allows you to modify the green box collider associated with the body of the vehicle.



Again, make use of the front and side views and use the slider bars to modify the green box collider.



The second and final section is the Vehicle Settings, this section enables you to modify parameters about the behaviour of the vehicle, such as suspension height, spring stiffness, traction and so on.

Unfortunately, it is not possible to see the result of changing these parameters within the Unity Editor, to see how changes to these values affects the vehicle, you need to Play the scene, and test drive your vehicle.

Please refer to the next section of this user guide for information on testing your vehicle.

Test Driving a Vehicle

Before you can test drive a vehicle, you need to add some game objects to your scene.

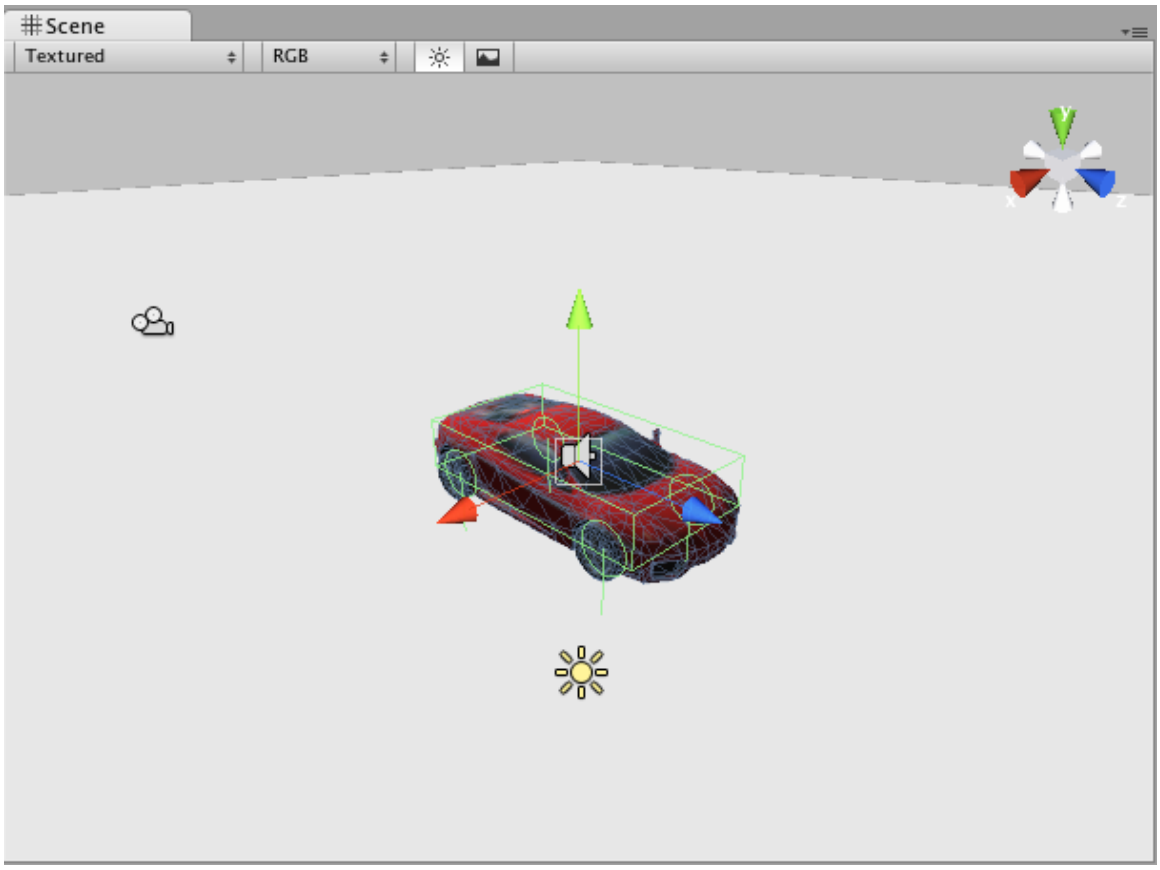
1. First, add a plane game object, click the GameObject menu, and select Create Other, Plane.

2. Next position the plane at 0, 0, 0, select the Plane game object in the Hierarchy panel, and enter 0's into the X, Y and Z position fields within the Inspector panel.

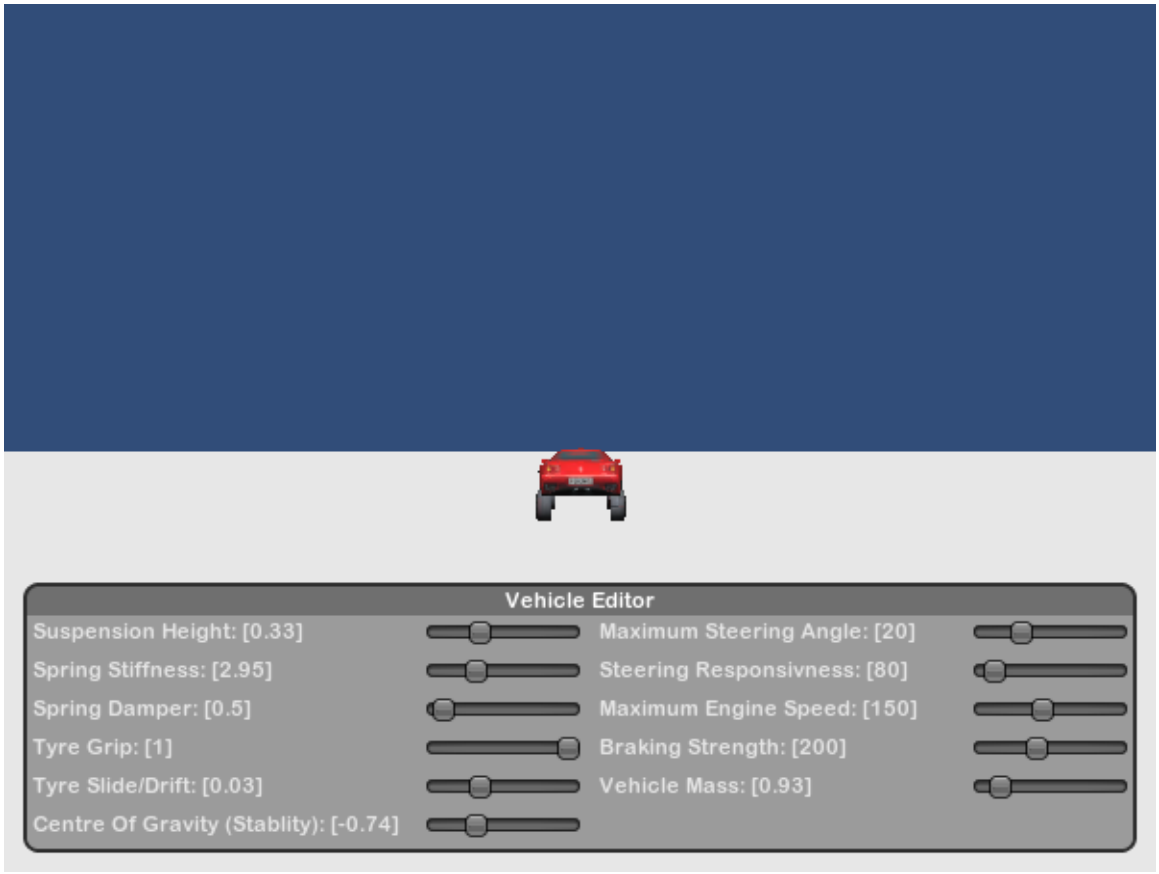
3. Now do the same for the VehicleRig, however position the vehicle at 0, 2, 0, so the vehicle is position slightly above the Plane game object.

4. Finally, change the size of the Plane, by entering 10, 1, 10, into the X, Y and Z scale fields within the Inspector panel. And add a single directional light, click the GameObject menu, and select Create Other, Directional Light, change its X rotation value to 45.

   TIP – If you now select the VehicleRig in the Hierarchy panel, and then move the mouse over the Scene view and press F on the keyboard, you should end up with the view below.



5. Now click the Play button, or click the Edit menu, and select Play, or simply press Ctrl+P.

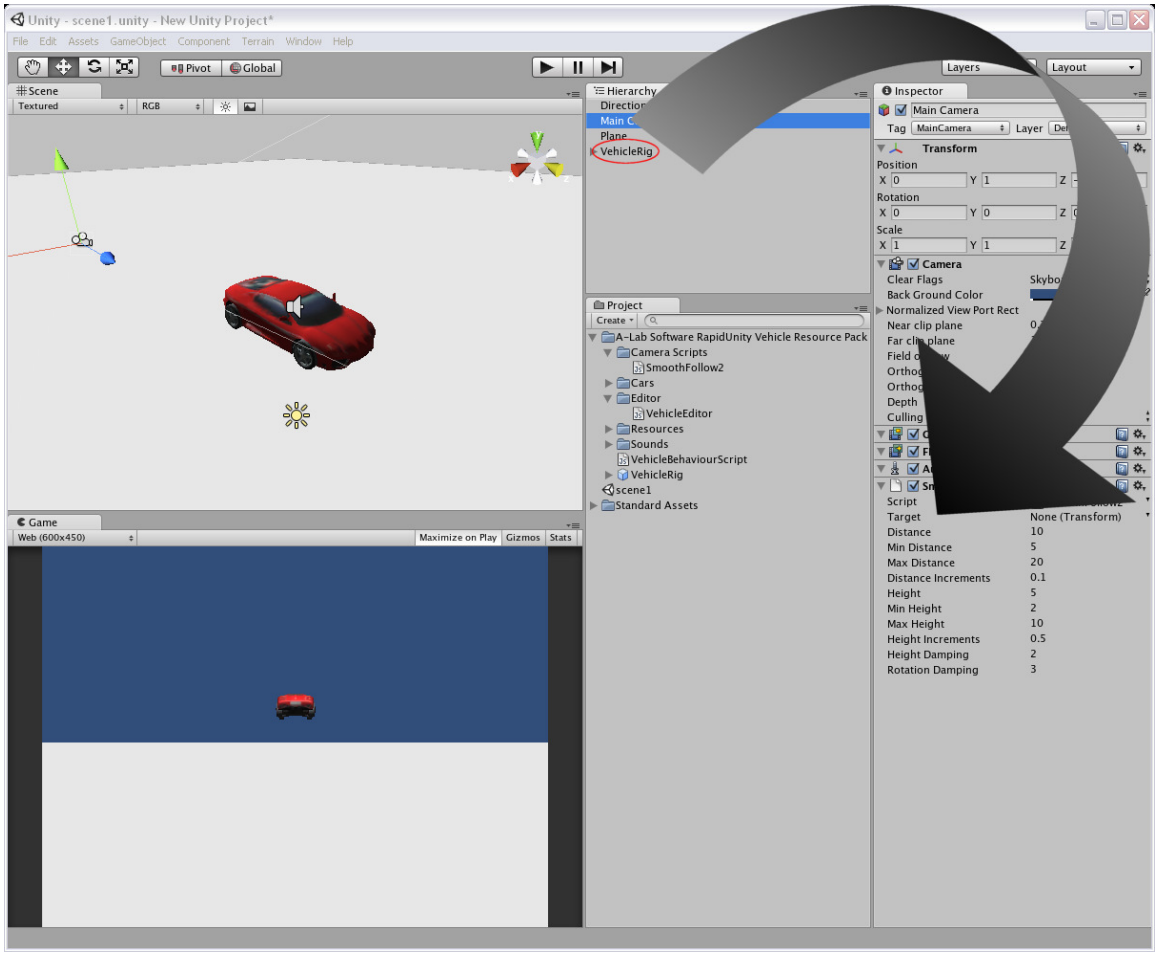


If you use the cursor keys you can drive the vehicle about on the Plane, and you will notice a nice grey panel with lots of sliders that you can use to affect the behaviour of your vehicle.

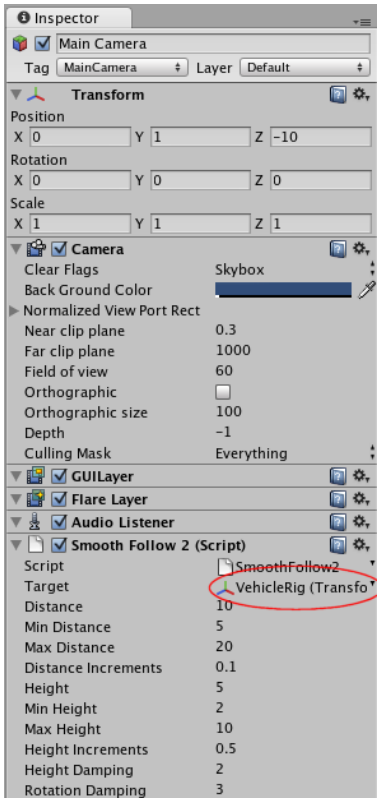
Hang on a minute, I cannot see the car very well, I want the camera to follow the car!

6. No problem, simply drag the SmoothFollow2 script from the Scripts folder in the Project panel to the Main Camera object in the Hierarchy panel.

7. One last step, select the Main Camera in the Hierarchy panel, and then click and drag the VehicleRig from the Hierarchy panel to the Target field within the Inspector panel.



You should end up with the VehicleRig transform assigned to the Target field.



Now try Playing the scene again, this time when you use the cursor keys the camera follows the vehicle around the scene.

The SmoothFollow2 script is a modified version the same script included with the Standard Assets, there are two extra features, you can use the + and – keys on your numeric keypad to move the camera closer or further away from the vehicle, and you can also use the Page Up and Page Down keys to move the camera up or down as it follows the vehicle, these keys allow you to find your preferred viewing angle.

Vehicle Editor Settings

As mentioned earlier they is a nice grey panel with lots of sliders that you can use to affect the behavior of your vehicle.

But what do they all mean? Below is an explanation of each parameter and how changes will affect the vehicle.

Suspension Height – This is fairly straightforward, by adjusting this slider you can set the ride height of your vehicle. A value of 0 means the vehicle has no suspension.

Spring Stiffness – This setting determines how hard or soft the suspension springs will be, use the slider to find the preferred setting. A higher value makes the suspension stiffer.

Spring Damper – This setting affects the suspension velocity. A larger value makes the suspension spring move slower.

Tyre Grip – This is an interesting setting, it is similar to Vehicle Drag, the lower the value, the slower the vehicle will accelerate, the higher the value the quick the vehicle will accelerate.

Tyre Slide/Drift – I like this setting, you can have lots of fun trying to find the value that best suits your vehicle, a value of 0.1 provides no tyre slide or drifting at all, and a value of 0.01 is provides plenty of drift and is great fun sliding your vehicle around a scene.

Centre of Gravity (Stablity) – This is probably the most important setting, and affects the handling of your vehicle the greatest. By adjusting this value, you can cause your vehicle to pull wheelies, or barrel roll over when turning sharply. Quite a bit of trial and error can be used with this setting to find the optimum value for your vehicle. As a guide look at this value for each of the included vehicles, you will notice there is quite a difference between the Jeep and the F360.

Maximum Steering Angle – This setting defines the limit of the steering, a lower value provides less steering, and a higher value allows for greater steering, and the ability to make sharper turns.

Maximum Engine Speed – This value limits the engine top speed, a lower value reduces the available top speed, and a higher value allows the vehicle to go faster.

Vehicle Mass – This is weight or volume of the vehicle, for this value I have tried to replicate the real world, so this value is the equivalent to kilogram's, so for instance the Subaru is set to 1600, however in real life a Subaru car is approx 1400kg. A higher value equates to a heavier vehicle.

Vehicle Drag – This setting affects the air resistance of your vehicle, for instance a vehicle such as a heavy truck with a large flat front will have a greater air resistance to a nice low sleek sports car, so you can use this setting to simulate the aerodynamic efficiency of your vehicle. A higher value will cause the vehicle to slow down faster when forwards motion (acceleration) is removed.

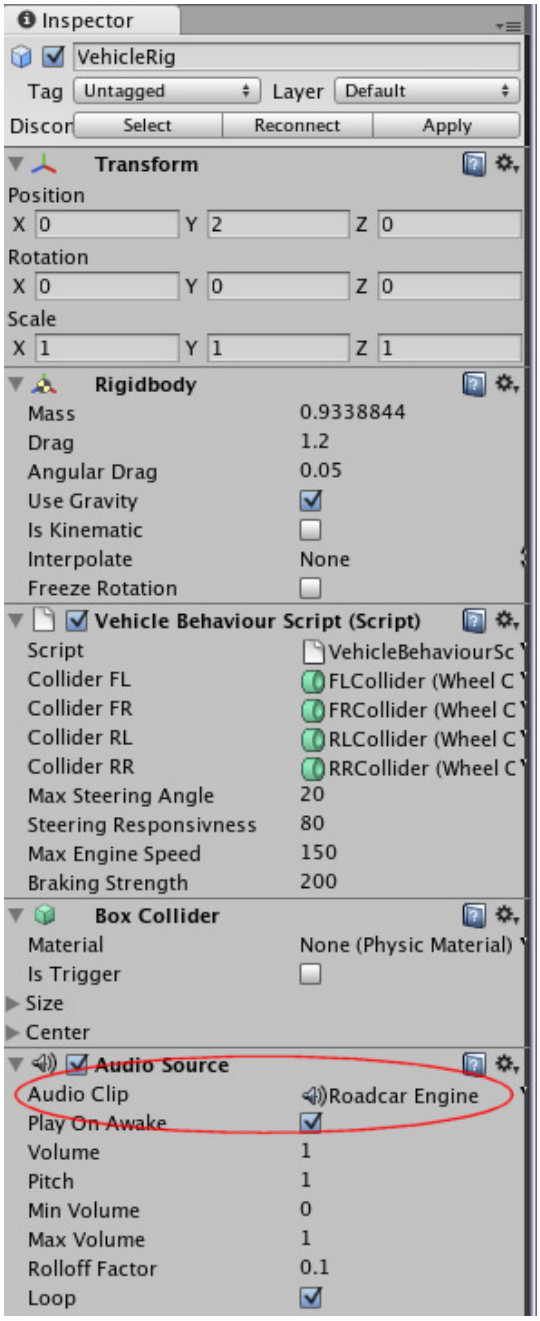
Additional Features

Something that I have not covered yet is the engine sound, included with the Vehicle Editor Resource Pack are 6 different engine sounds.

These can be easierly changed, and you can also add your own.

How to change the engine sound

1. Select the VehicleRig within the Hierarchy panel, in the Inspector panel you will now see all the components of the VehicleRig, at the bottom of the panel is Audio Source.



2. To change the engine sound, simply click the Audio Clip dropdown and select a different engine sound.

How to add your own engine sounds

1. To add your own engine sounds, simply copy an audio format file into the \Assets\A-Lab Software RapidUnity Vehicle Resource Pack\Sounds folder within the Project.

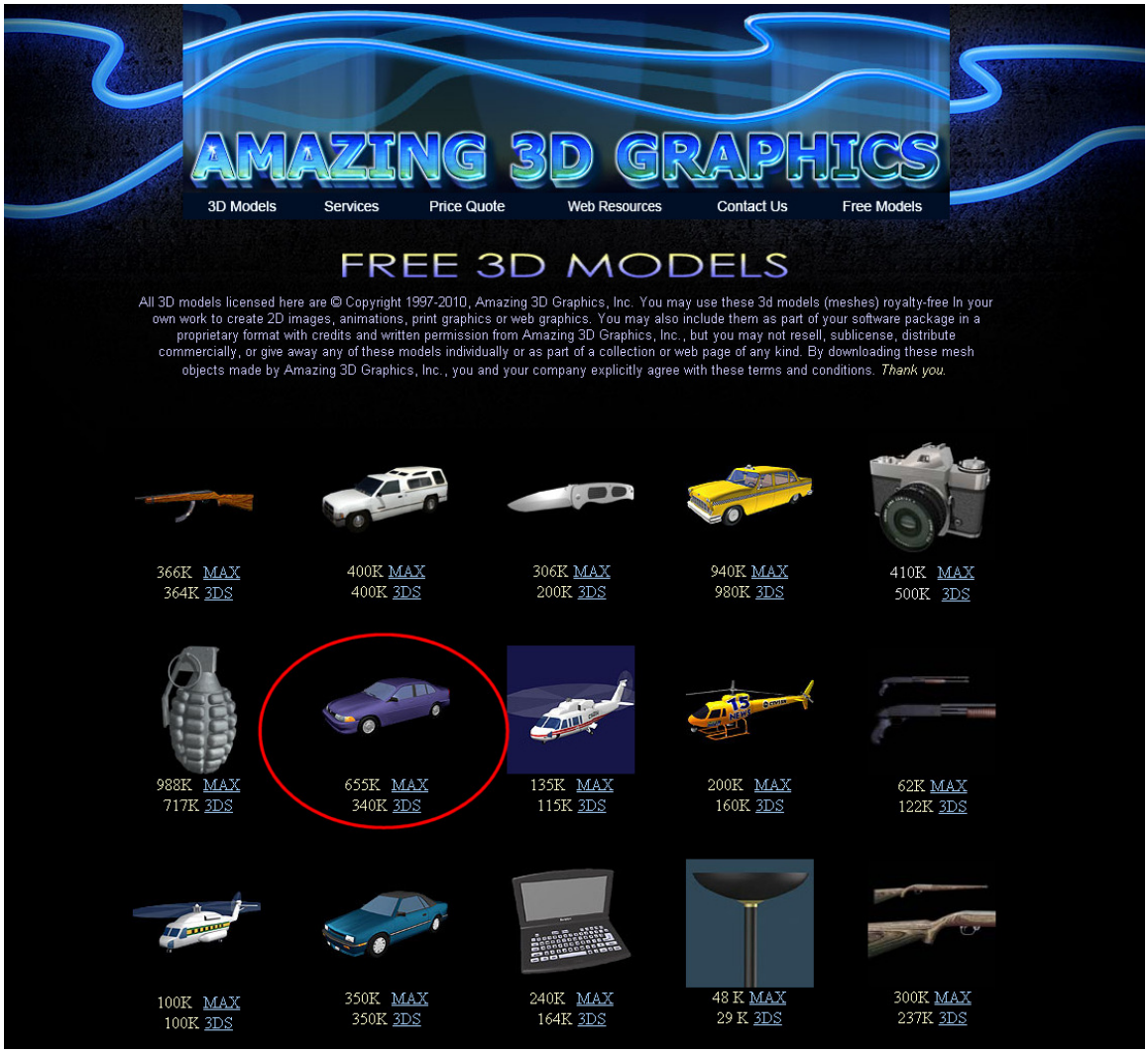
2. Then when you click the Audio Clip dropdown, your newly added audio file will appear.

How to add your own custom vehicle bodies and wheels

Adding your own vehicle bodies and wheels is simple, and this is one of the most powerful features of the Vehicle Editor Resource Pack.

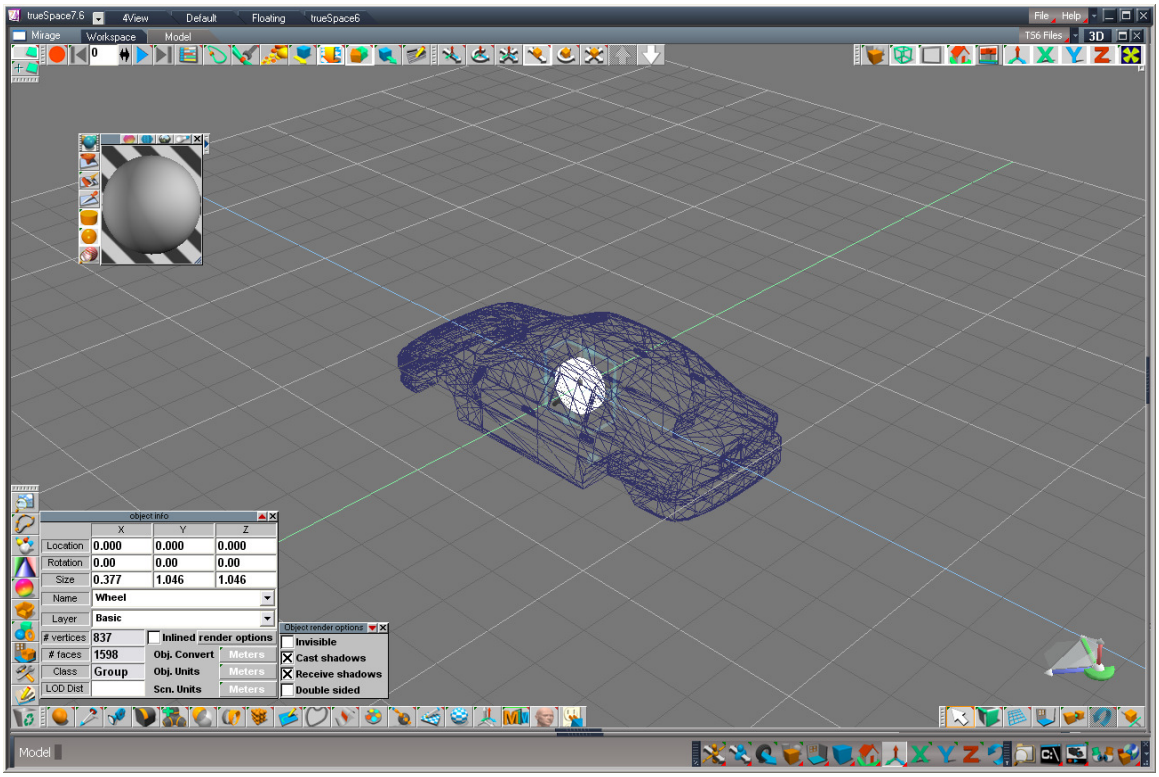
In the steps below, I will take you from downloading a new 3D car model from the internet to importing into Unity, and adding to the Vehicle Rig.

1. First you need a 3D car model, visit the following website http://www.amazing3d.com/free/free.html, and download the Mirage 3ds file by clicking the 3DS link below the car image.



2. Once downloaded, extract the contents of the zip file to a folder on your computer.

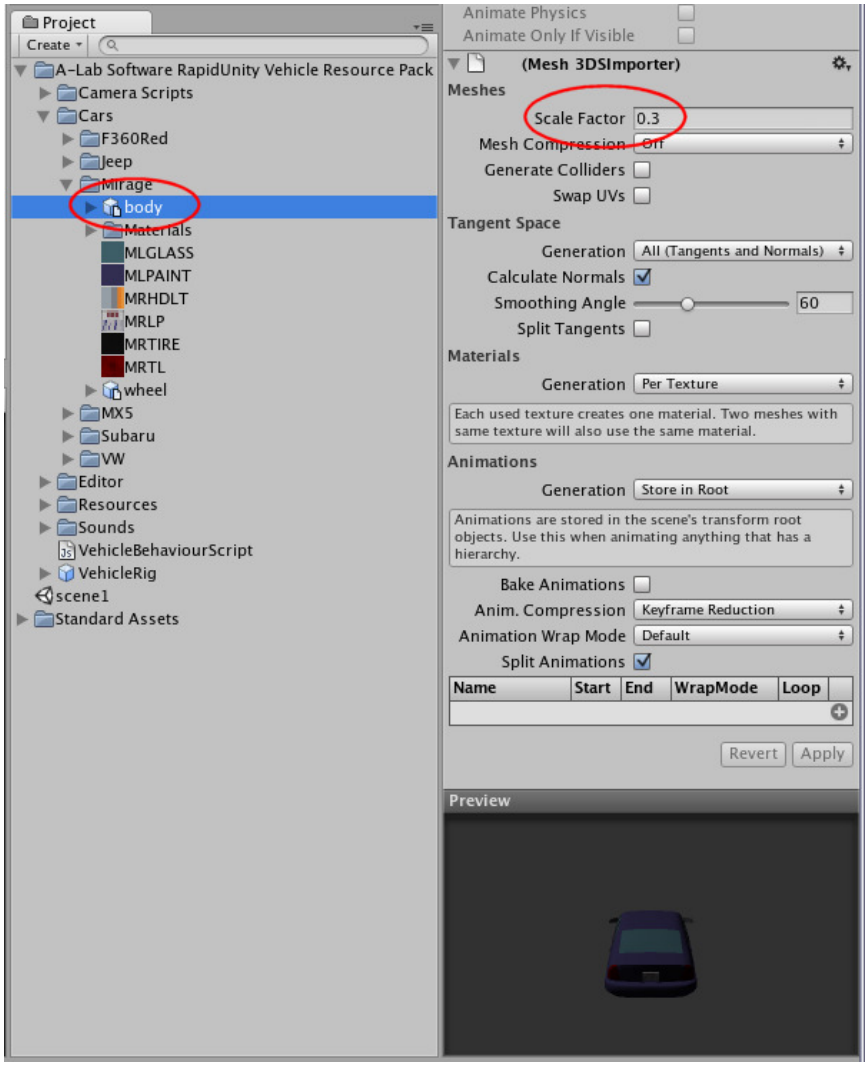
3. Now using your favourite 3D application (mine is trueSpace7.6, which is available for FREE, and can be downloaded from http://www.caligari.com/downloads.html) separate the wheels and body meshes and center them at 0, 0, 0, and set the axis point to the center of the body and wheel meshes respectively.



4. Now save the body as a 3ds file, name it body.3ds and save it in a new sub-folder of \Assets\A-Lab Software RapidUnity Vehicle Resource Pack\Cars and call the folder Mirage.

5. Do the same for the wheel mesh, note you only need to save a single wheel.3ds file.

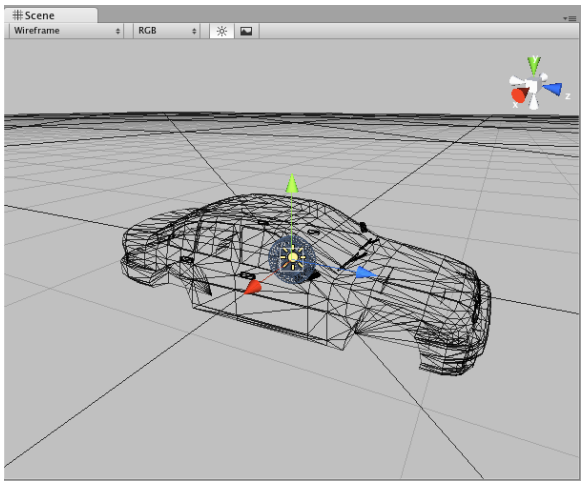
6. Now open the project in Unity with the VehicleRig in it, then select the body object within the Project panel, under A-Lab Software RapidUnity Vehicle Resource Pack\Cars\Mirage.



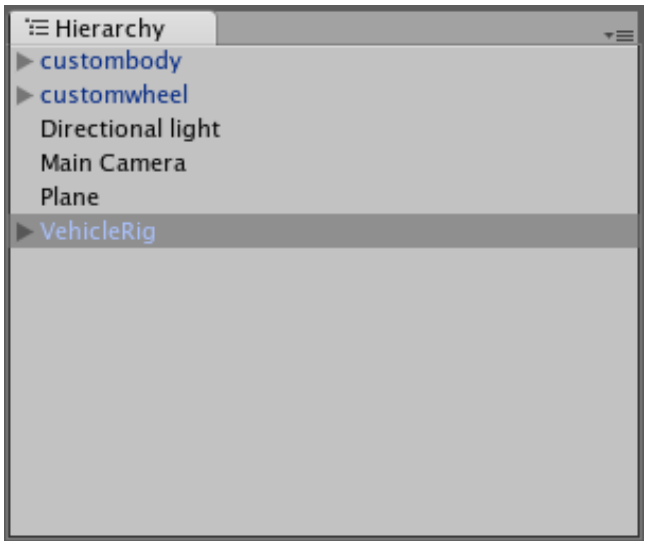
7. In the Inspector panel, change the Scale Factor from 0.1 to 0.3, and drag the body mesh into the Scene view, and position at 0, 0, 0.

   NOTE: When importing other vehicle models, you may need to use different Scale Factor values.

8. Repeat the above steps for the wheel mesh, and your view in Unity should look like the image below.

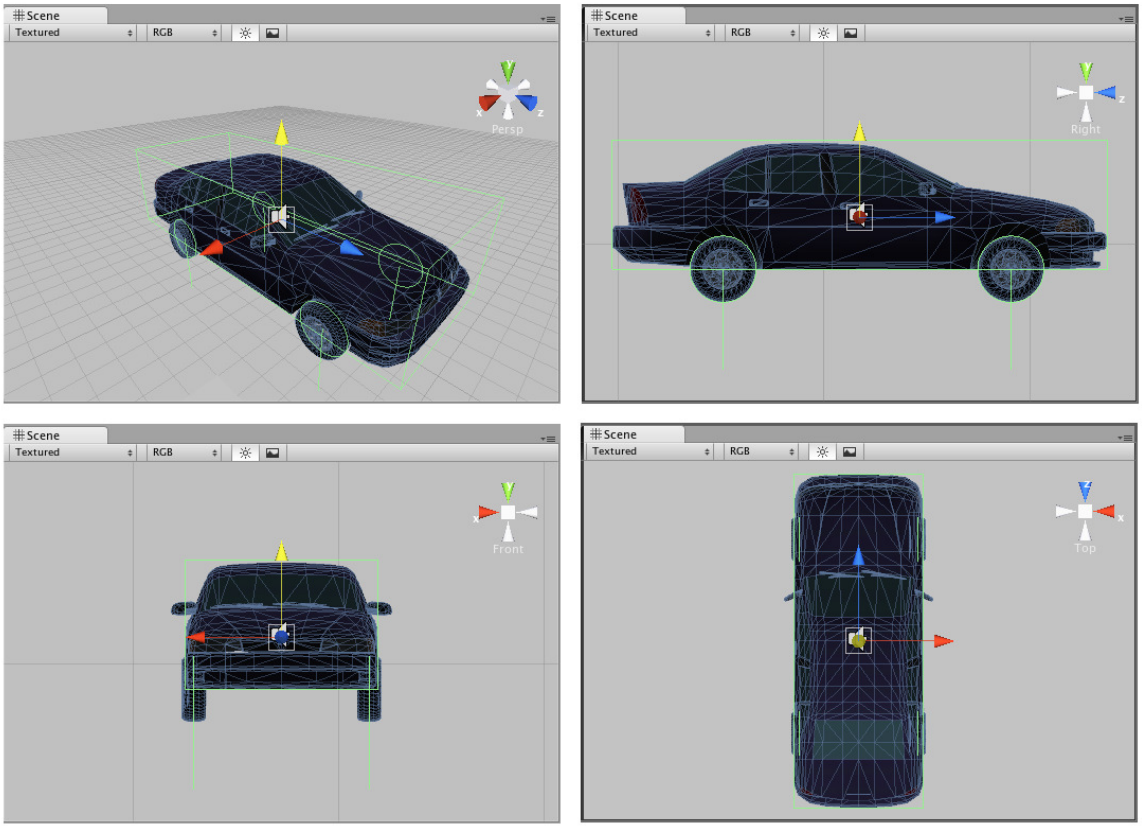


9. Finally, rename the body and wheel game objects in the Hierarchy panel to custombody, and customwheel.



10. And now to switch the current VehicleRig body and wheel meshes for your custom meshes. Select the VehicleRig in the Hierarchy panel, and then click the Custom Body and the Custom Wheel buttons in the Vehicle Editor panel. Your VehicleRig should now have your custom body and wheels added.

NOTE: You may need to change the green box and wheel collider sizes, and wheel positions. Please refer the section entitled **How to use the Rapid Unity Vehicle Resource Pack in Unity** of this user guide, steps 7 and 8 for information on how to adjust the colliders and wheel positions using sliderbars.



The image above shows the result after using the Vehicle Editor slider controls.

New Features in v1.5

With v1.5 of the Vehicle Editor I have added compatibility with the official **Unity Car Tutorial** car set-ups.
http://unity3d.com/support/resources/tutorials/car-tutorial

This amazing tutorial includes two very good car rig's and even better control scripts, one car is an arcade type racer, and the other is a high-end racing simulation type setup.

Both car rig's have several parameters that can be modified to affect the cars handling, in fact the high-end simulation rig even has a parameter which defines the time it takes from when the handbrake is released, to the physical time it takes for the mechanical operation of the brake pads being disengaged from the wheels! This is a serious car simulation setup!

And v1.5 of my Vehicle Editor allows you to modify any of these parameters both from within the Unity Editor, or at runtime, and even better you can save and load complete car settings to and from file, using my new file format .ves, **V**ehicle **E**ditor **S**ettings.

The following sections explain how to make use the new version of the Vehicle Editor with the Unity Car Tutorial car set-ups.

Using the Vehicle Editor with the Unity Car Tutorial

There are two ways to use of the Unity Car Tutorial car set-ups, firstly you can simply install the Vehicle Editor within the Unity Car Tutorial project, or secondly you can add the Vehicle Editor and Unity Car Tutorial car set-ups to a brand new project.

Vehicle Editor within the Unity Car Tutorial Project

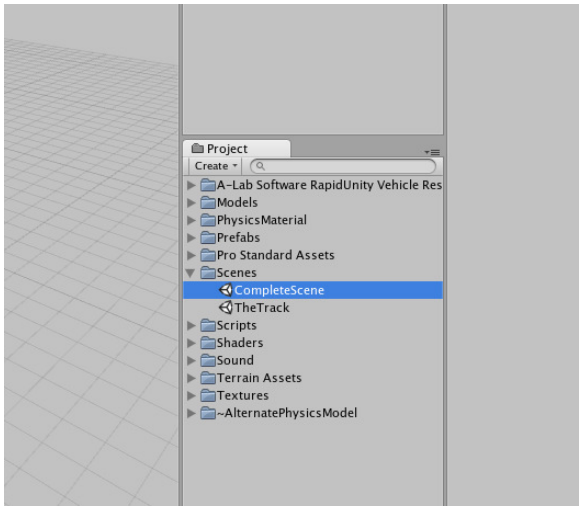
1. First download and extract the CarTutorial.zip file from the following link:

http://unity3d.com/support/resources/tutorials/car-tutorial

2. Next open the CarTutorial project within Unity.

3. Now install the RapidUnity Vehicle Editor Resource Pack, refer to the section entitled Installation within this document.

4. Now open the CompleteScene under the Scenes folder in the Project panel.

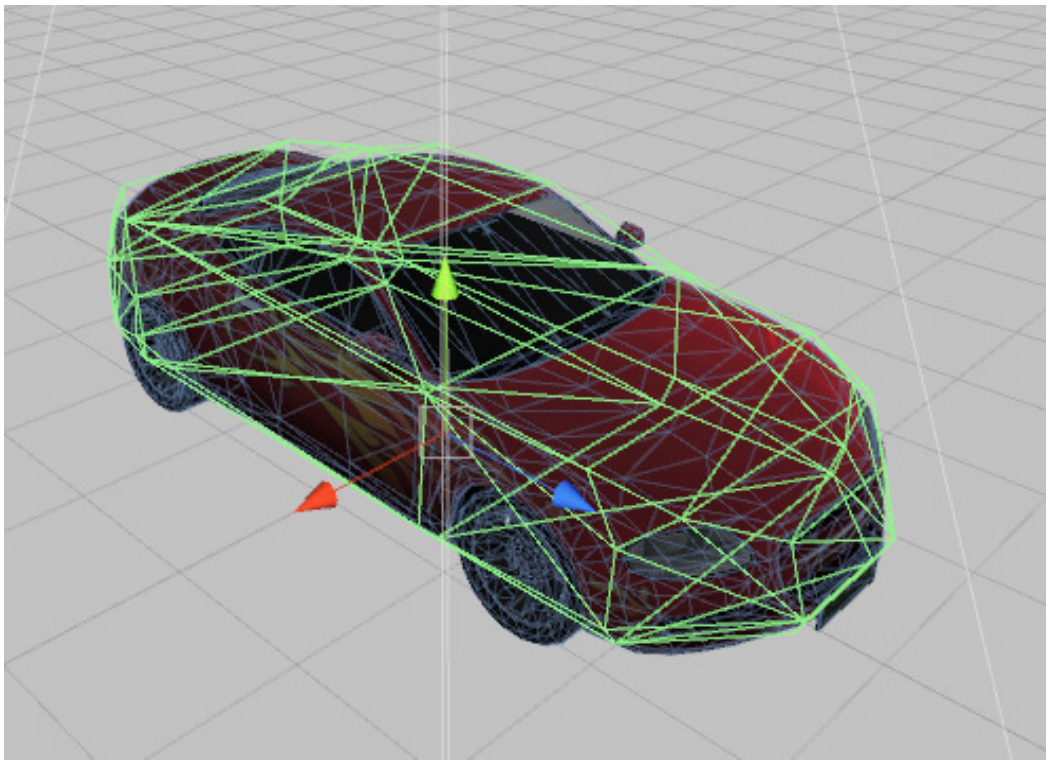


5. Once the scene is open, you will see lots of items in the Hierarchy panel, however nothing is displayed in the Scene view, click on the Car in the Hierarchy panel, and then move the mouse over the Scene view and press F on the keyboard, to move the view to focus on the Car game object.

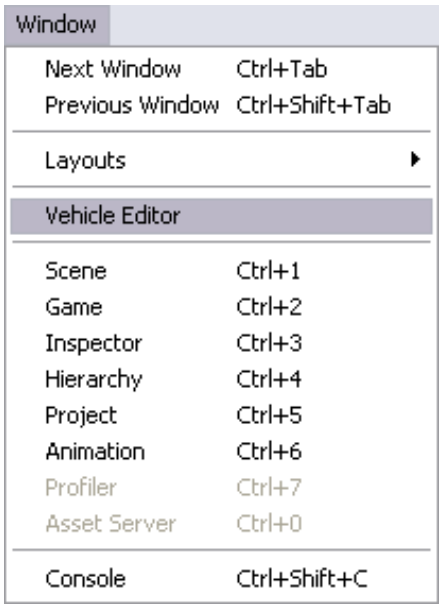
6. Now make a note of the Car game object position.

7. Now change the Car game object position and rotation to 0, 0, 0, and then move the mouse over the Scene view and press F on the keyboard again.

   Note: You might need to click on the Car game object in the Hierarchy panel before moving the mouse over the Scene view.



8. Next, you need to open the Vehicle Editor, to do this simply click the Window menu, and select Vehicle Editor.



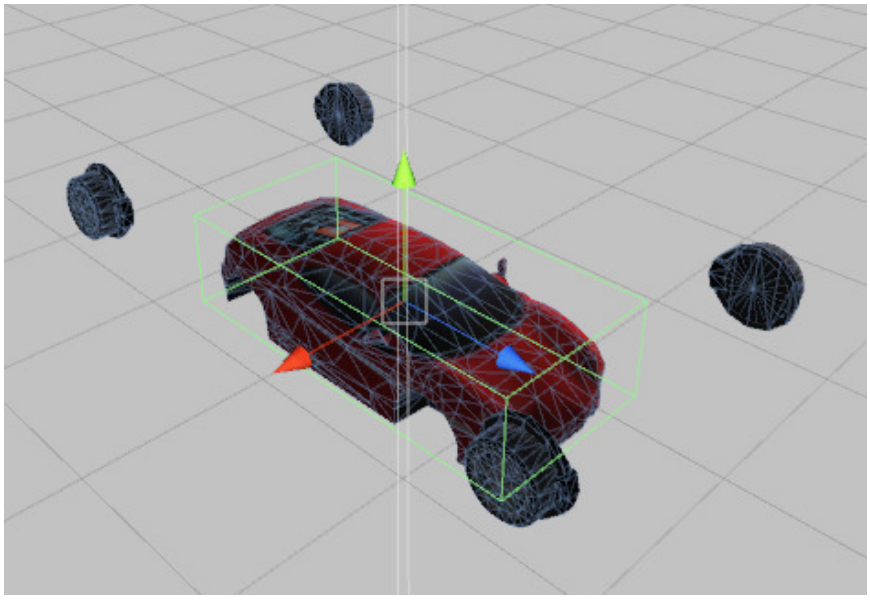
9. Now the Vehicle Editor panel will be displayed.

10. If you have used the Vehicle Editor before, you will now that the Body Settings section contains sliders to adjust the box collider associated with the selected VehicleRig. However because the Default Unity Car is selected the Vehicle Editor detects that this rig uses a mesh collider and not a box collider. If you are going to be changing the body and wheels of this rig, it is recommended you change the mesh collider for a box collider, simply click the Yes please! button under the Body Settings section of the Vehicle Editor.

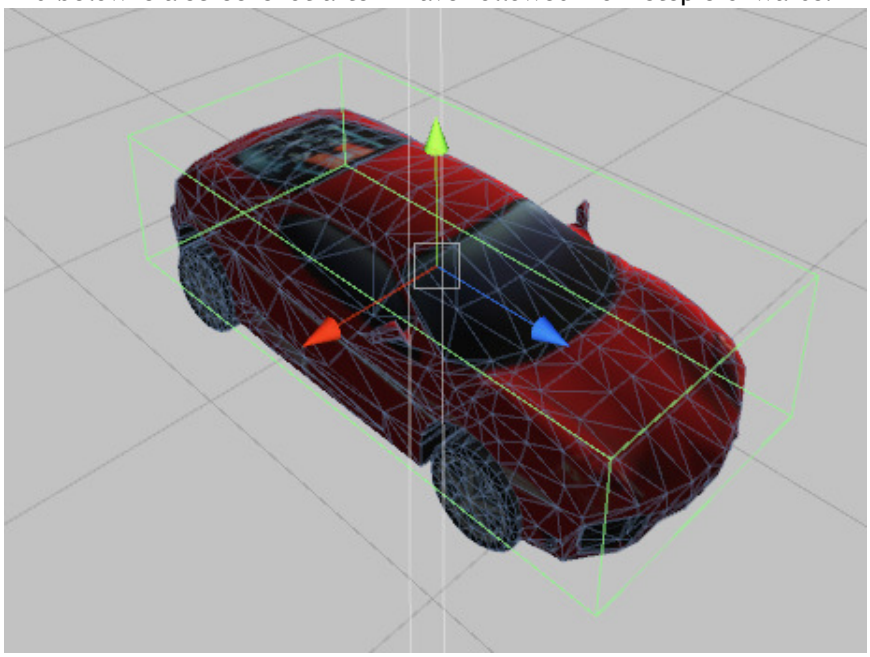
11. You will now notice that the Body Settings section contains the sliders to adjust the collider associated with the rig, and that the Car game object in the Scene view now has a small green box as its collider.

12. Now refer to step 5 of the section entitled **How to use the Rapid Unity Vehicle Resource Pack in Unity** within this document to learn how to change the car wheels and body, and to adjust the collider size, and wheel positions.

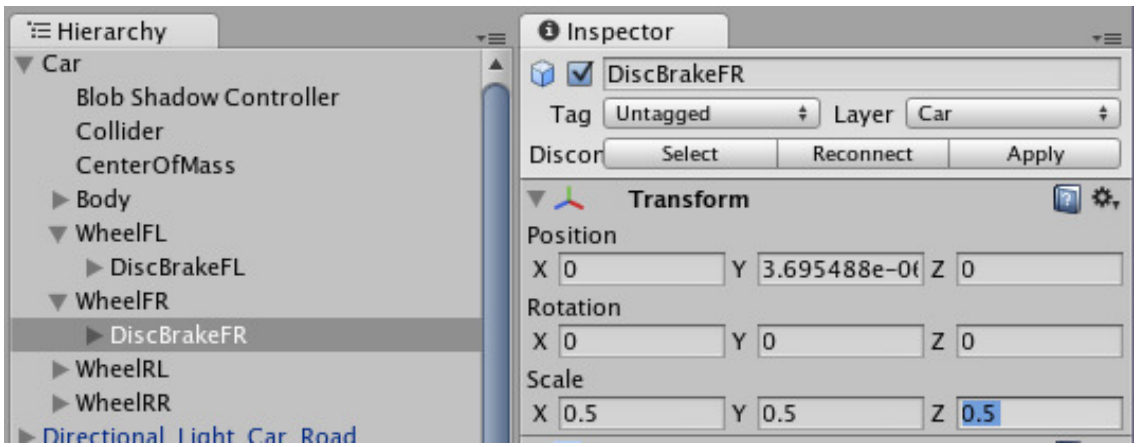
    Below is a screenshot of the Default Unity Car with the Ferrari wheels and body added.



    And below is a screenshot after I have followed from step 5 onwards.

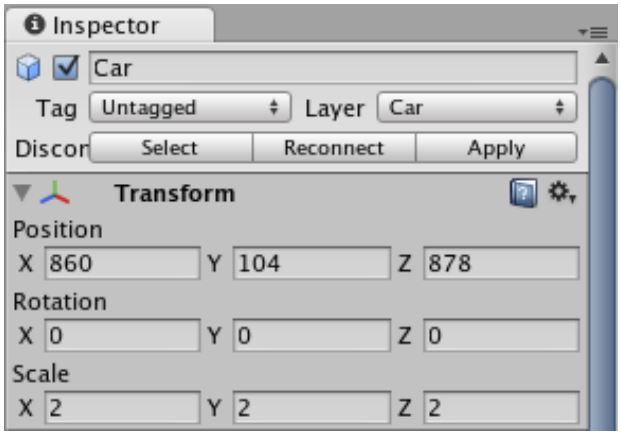


13. One thing you might have noticed with the Default Unity Car is that the brake disc game objects are still visible, due to the scale of the included Ferrari model, you need to either scale the brake discs down, or simply hide them. Simply drill down into the Car game object structure within the Hierachy panel, and select each DiscBrake game object and change the Scale to 0.5 for x, y and z, or uncheck the tick mark next to the game object name in the Inspector panel to hide the brake disc.



Note: Unfortunately, because the wheel meshes are child objects of the brake discs, the wheels have also been scaled down, no problem simply click the Ferrari wheel button again in the Vehicle Editor to re-add the wheels.

14. Finally change the Car game object position back to the values you recorded in step 6 (you might need to move the car up the y axis slightly), and change the Scale to 2 for x, y and z, then Play the scene!



15. Now play with the Car in the Game view and drive it about, you can use the Runtime Vehicle Editor panel to adjust the different parameters that affect the vehicles behaviour, the first parameter you might want to adjust in the Wheel Radius, this can be found by scrolling to the bottom of the sliders, you may also want to adjust the Center of Gravity as well if the car rolls over too easierly. Have fun, playing with all the available settings trying to find your desired setup. And don't forget to read the section entitled **Saving and Load Vehicle Editor Settings**, which explains how to Save your settings at runtime and apply them with the Unity editor.

Using the Vehicle Editor and Unity Car Tutorial car set-ups in a New Project

1. Before you can use the Unity Car Tutorial car setups in a new project with the Vehicle Editor, you first need to export the car setups as UnityPackages.

2. Open the Unity Car Tutorial project, and open scene CompleteScene, then follow the instructions on page 9 of section 2 of the Unity Car Tutorial documentation, this pdf can be downloaded from the following link:
   http://unity3d.com/support/resources/files/CarTutorialSection_2.pdf

3. On page 11 of the instructions name the package UnityDefaultCar and click Save.

4. Once the export completes, repeat the above steps this time selecting the Prefabs folder under \~AlternatePhysicsModel within the Project panel, and name the package UnityAdvancedCars.

5. Now that we have both the car set-ups as Unity Packages, we can create a New Project, and import the packages. So click File, New Project.

6. Next click the Asset menu, and select Import Package…, select the A-Lab_Software_RapidUnity_Vehicle_Resource_Pack_v1_5.unitypackage file and click Open.

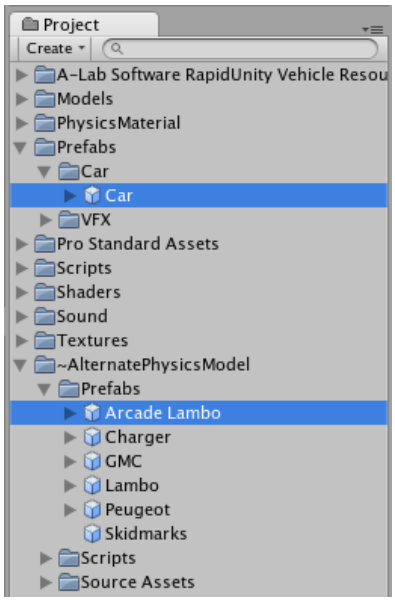
7. Next import both the UnityDefaultCar.unitypackage and the UnityAdvancedCar.unitypackage packages.

8. And finally, now import both the UnityDefaultCarScript.unitypackage and the UnityAdvancedCarScript.unitypackage packages.

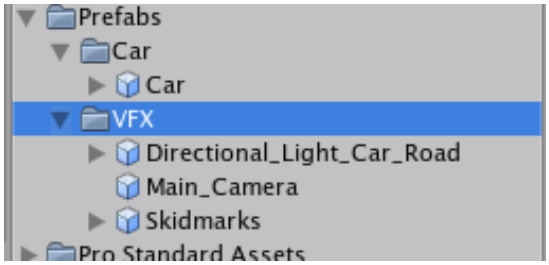
   These two packages contain modified versions of the Unity Car Tutorial control scripts, which includes the Runtime Vehicle Editor user interface.

9. That's it! You have everything you need in a nice clean new Unity project.

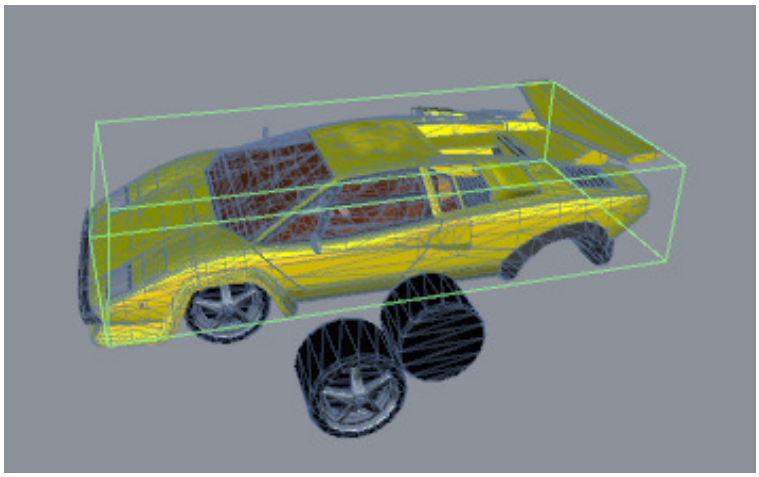
10. To use either of the new car set-up prefabs, simply drag either the Car or advanced car prefabs into your scene, and position at 0, 0, 0, so that you can use the Vehicle Editor to change the wheels and body, and position the wheels correctly. Please refer to step 5 of the section entitled **How to use the Rapid Unity Vehicle Resource Pack in Unity** within this document.



NOTE: When using the default car, you also need to drag the Skidmarks prefab into the scene, it is advised you replace the scene camera, with the Main_Camera prefab, finally add a light to the scene, otherwise when you run the project, the car body will not be visible.



IMPORTANT KNOWN ROTATION BUG: You may have noticed, that I have not changed the prefab rotation when dragging either the Car or advanced car prefabs into the scene. This is due to a **KNOWN BUG** with the Unity Car Tutorial car set-ups. If you rotate the prefab on its y-axis, and Play the scene, the wheels will become corrupt and change position, after you stop playing the scene, the wheels remain in the wrong position. See example below:



However, fear not I have discovered a rather bizarre trick to stop this from happening when you change the rotation of the car-setups in your scene.

1. Change the y-axis rotation to your desired value.
2. Save the scene.
3. Play the scene, and you will see the problem occur with the corrupt wheel positions.
4. Now stop the scene, and open the saved scene from step 2.
5. Now Play the scene again, this time you will notice the bug does not occur, hurray!

Saving and Load Vehicle Editor Settings

In this section I explain how to load and save my new file format .ves, **V**ehicle **E**ditor **S**ettings.

This is a really powerful feature of the Vehicle Editor, it enables you to save modified settings at runtime, and then apply them to the car set-up within the Unity editor.

You can use these Vehicle Editor Settings files with different Unity projects, and I am hoping to create a set to settings files for different types of vehicles, such as rally car, road car, racing car, drift car etc. I am also very interested to see what users come up with, and to try out their vehicle settings.

1. It is incredibly easy to save the vehicle settings, run your project, and when you wish to save the settings you have selected via the slider controls, simply click the Save Settings… button at the top of the Runtime Vehicle Editor dialog.

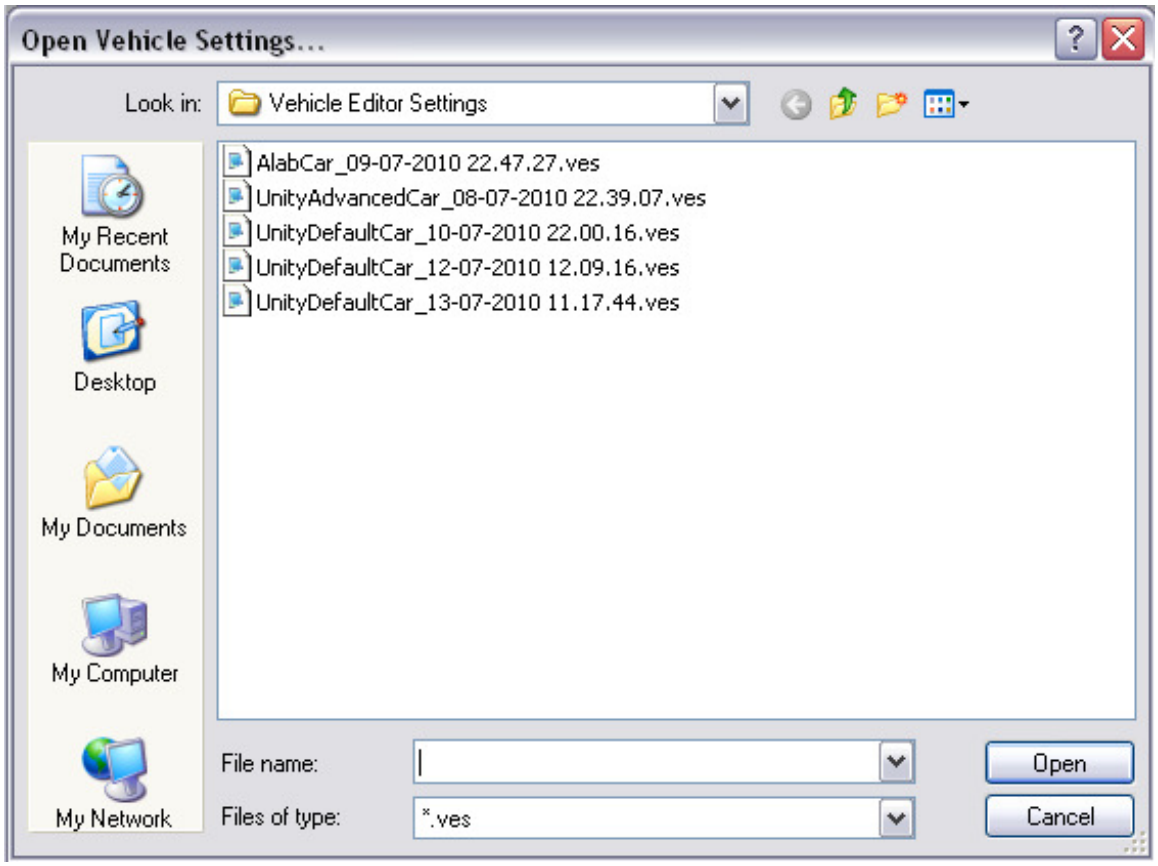
2. This will generate a file in the \A-Lab Software RapidUnity Vehicle Resource Pack v1.5\Vehicle Editor Settings folder named UnityDefaultCar_dd_MM_yyyy hh.mm.ss.ves or UnityAdvancedCar_dd_MM_yyyy hh.mm.ss.ves depending on the car prefab you are using.

3. You are welcome to rename the file to something more meaningful, however I would advise leaving the prefix of UnityDefaultCar_ or UnityAdvancedCar_ or AlabCar_ so you know which vehicle rig the ves file is for. However the Vehicle Editor is clever enough to display an error message if you try to apply a ves file to the wrong vehicle rig.

4. Ok, now onto loading the saved settings file within the Unity editor. Stop your project, and goto the Vehicle Editor, at the top of the dialog panel is a button entitled Load Settings…, click this button.

5. A file dialog is now displayed, find the folder \A-Lab Software RapidUnity Vehicle Resource Pack v1.5\Vehicle Editor Settings, and select the file you saved, and click Open.

   Tip: The file will be prefixed with the type of rig you are using, and the date and time is appended to the end of the filename, e.g UnityDefaultCar_10-07-2010 22.00.16.ves.



6. The vehicle settings saved in the file will now be applied to the vehicle within the Unity editor, you can check the Vehicle Editor sliders and values to see the new settings have been changed.