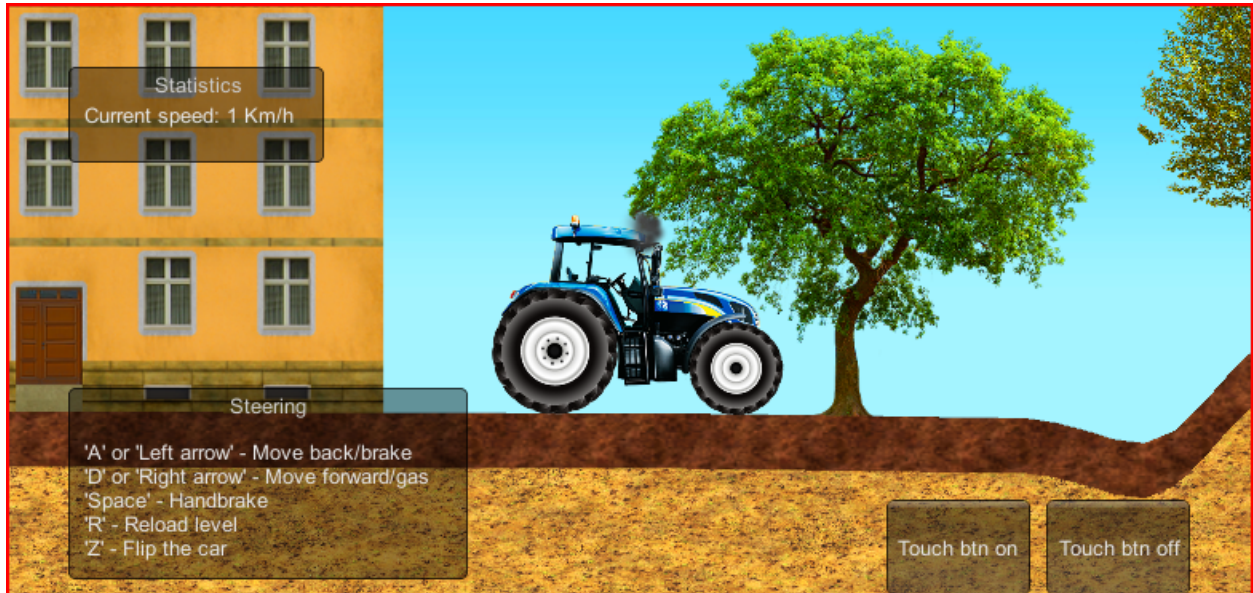


2D VEHICLE KIT 1.1

Basic manual



Overview

This asset allows you to easily implement 2D vehicle in your game. It's extremely easy to setup and tune - just in several clicks you will implement it anywhere. It's really fast and performance safe - ideal for smartphones.

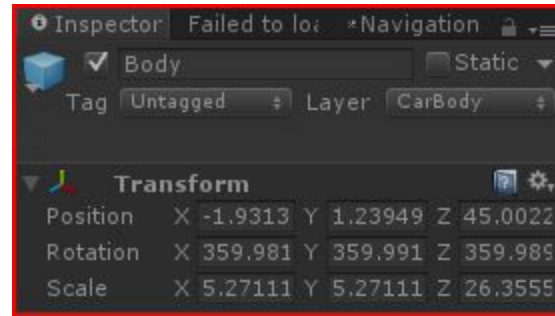
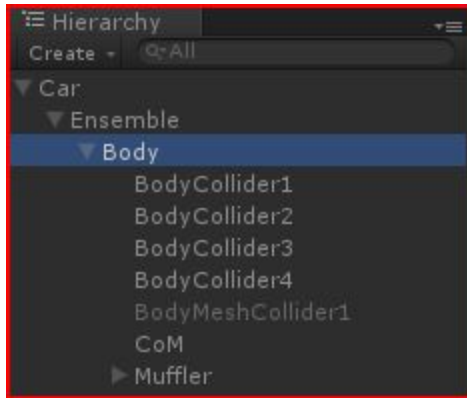
This asset works on all platforms supported by Unity3D.

Features:

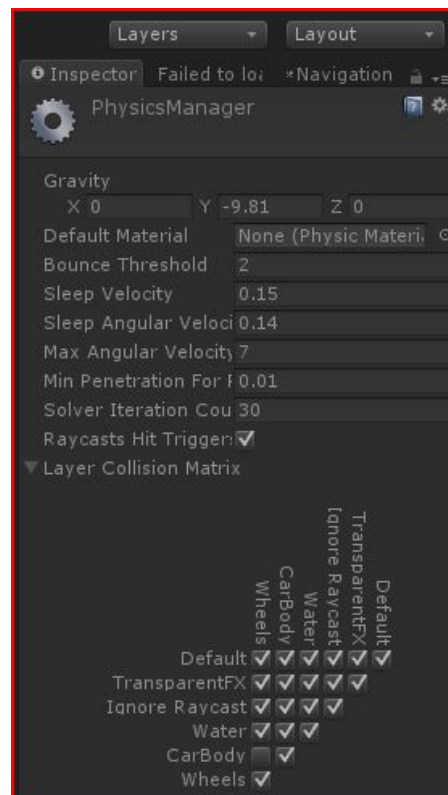
- Mobile friendly
- Keyboard/Touch controls
- Implemented basic vehicle sound system
- Implemented physics (rigidbodies, colliders, joints)
- 2 type of vehicles included (tractor, monster truck)
- Easily create vehicles with different number of wheels
- Support for Unity's new 2D physics (WheelJoint 2D , Rigidbody2D, Circle Collider 2D etc.)
- Check for game over when vehicle is on the roof for specified amount of time

How to use

1. Import the package to the project.
2. Create two new layers: "**CarBody**" and "**Wheels**". Select "**Body**" object from the project manager and set the layer in the inspector to "**CarBody**". Objects "**FrontWheel**" and "**RearWheel**" should be on "**Wheels**" layer.



3. Go to "Edit->Project settings->Physics" and unselect "CarBody/Wheels" checkbox under "Layer Collision Matrix". "Solver Iteration Count" set to "30" to make vehicle physics computation more accurate.

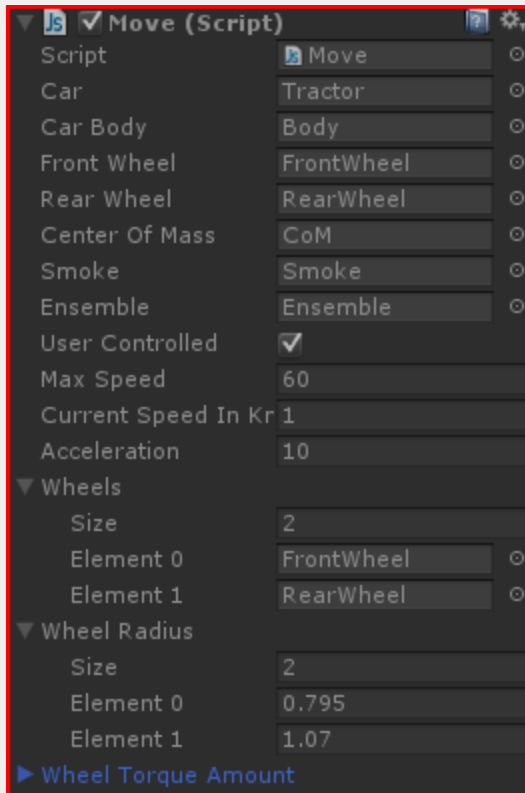


4. Load Demo-01 scene.
5. "Main Camera" object: set "Projection" to "Orthographic".

Scripts

Move.js

This script has options responsible for controlling the vehicle.



- **User controlled** - uncheck if you don't want the vehicle to be controlled by the user input
- **Max Speed** - set the maximum speed of your vehicle
- **Acceleration** - set some...
- **Wheels** - if you add more wheels to the vehicle just drag and drop them here
- **Wheel radius** - this variable will be set automatically
- **Wheel Torque Amount** - this variable will be set automatically

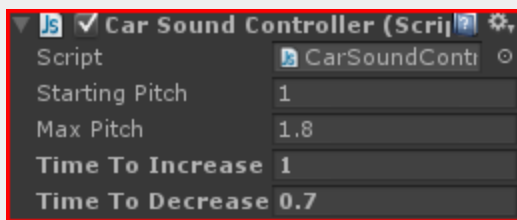
Gui.js

Provides some basic touch control buttons and info boxes. You can disable/enable them here and also setup custom textures for the buttons.



CarSoundController.js

Provides very basic sound system for the vehicle.

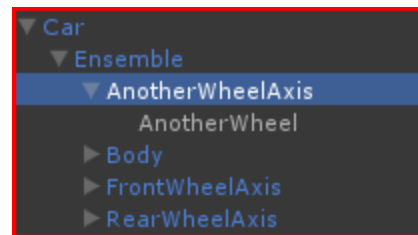
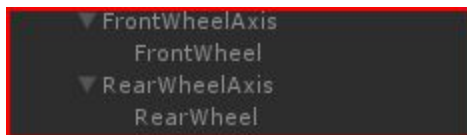


- **Starting Pitch** - base pitch for the sound sample
- **Max Pitch** - maximum pitch sound sample can reach
- **Time To Increase** - time to sample reach it's maximum pitch
- **Time To Decrease** - time to sample reach it's starting pitch

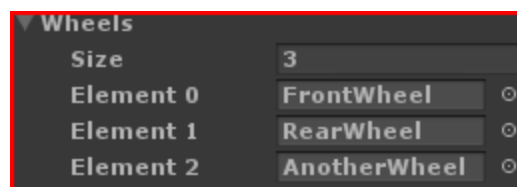
FAQ

1. How to add another wheel?

To add additional wheels (third, fourth) to the vehicle you need to select existing axis with the wheel attached (eg. FrontWheelAxis) in the 'Hierarchy' window and duplicate it (ctrl+D). Then just rename it to eg. "AnotherWheelAxis" for new axis and "AnotherWheel" for new wheel.



Next select newly created wheel and drag it onto Wheels section in the Move.js script attached to the vehicle object. This step is required if you want the torque force was applied to the wheel.

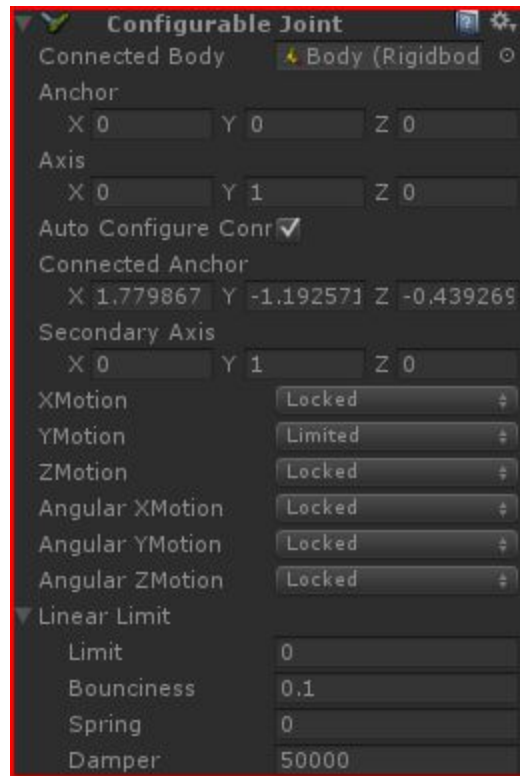


2. How to change the position of the wheel, body?

Just select the sprite and drag it to the position you want.

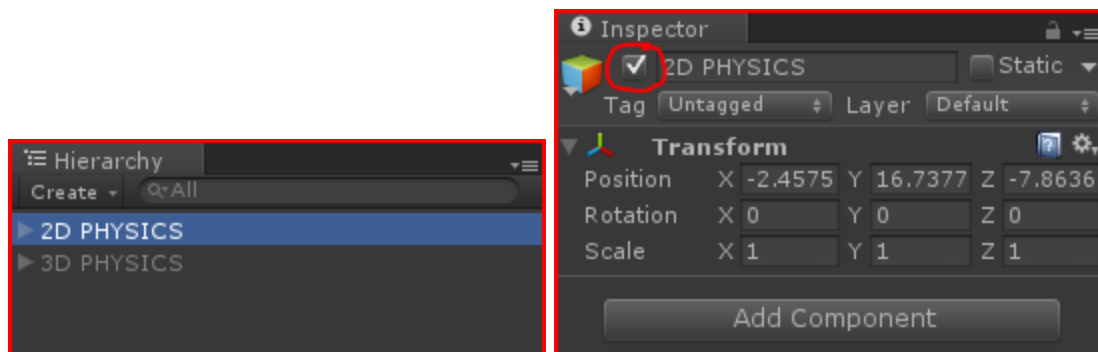
3. Where to change the suspension behavior?

Go to the 'Configurable joint' component attached to each wheel axis and tune the 'Linear limit' section ('Limit', 'Bounciness', 'Spring', 'Damper'). For 'Bounciness' don't set the value larger than 1.0 to avoid strange behavior.



Update for 2D physics

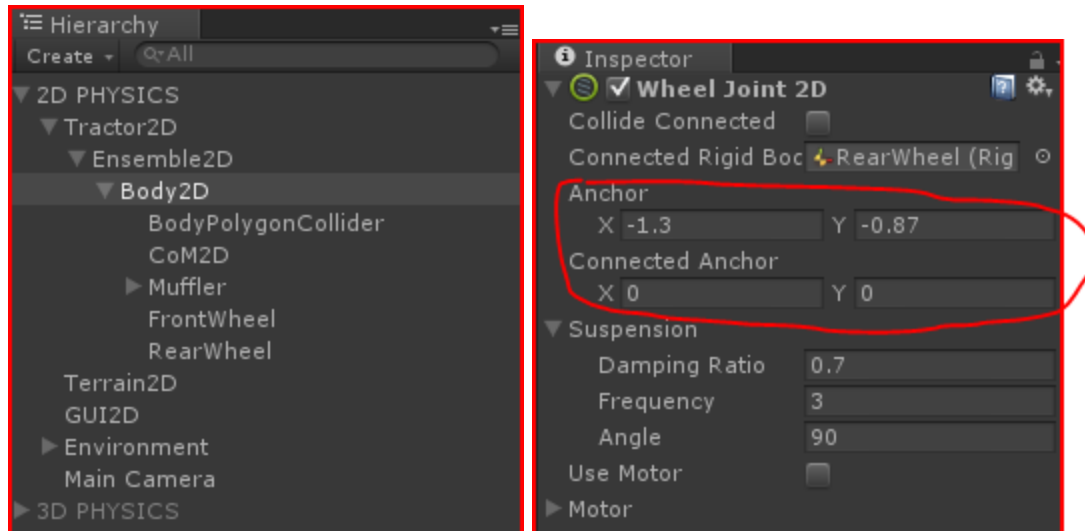
2D Vehicle Kit now supports Unity's 2D Physics. To choose between 2D and 3D physics you need just select the mode you want in the 'Hierarchy' panel and then enable it in the inspector as shown below:



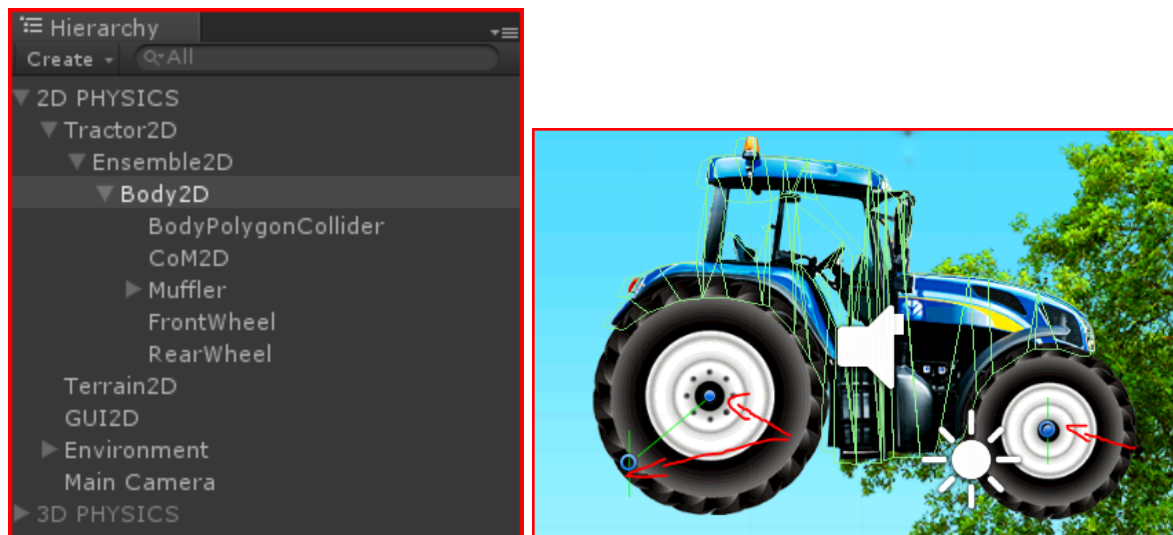
FAQ

1. How to change the position of the wheel, body?

Change the anchor points as shown below

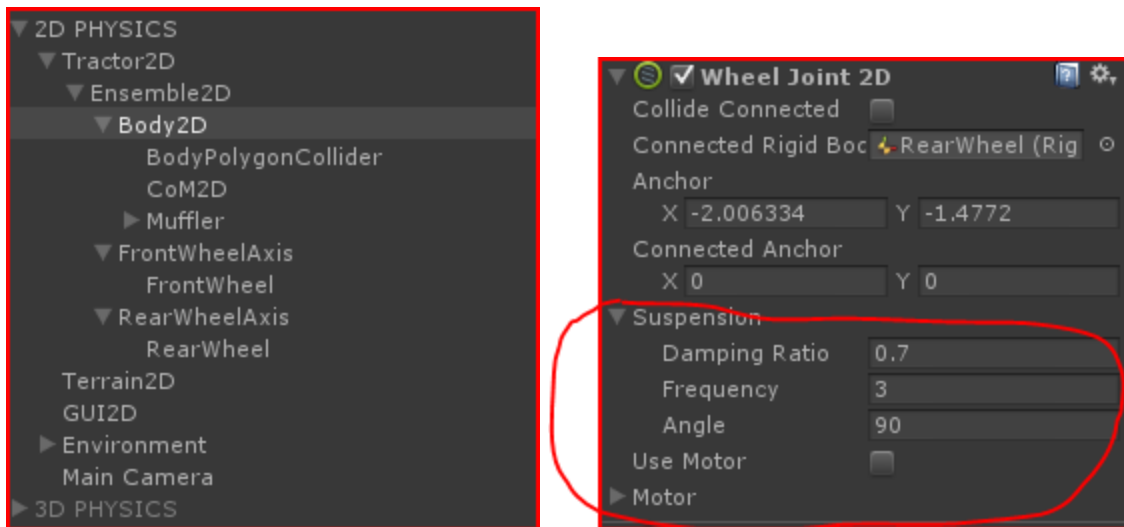


or just drag the 'Wheel Joint 2D' points to the place you want



2. Where to change the suspension behavior?

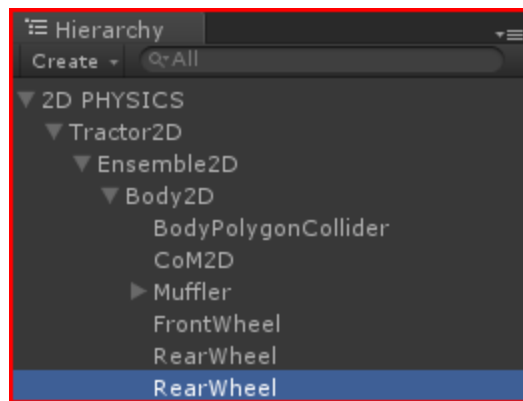
You can tweak such parameters as “Damping ratio”, “Frequency” and “Angle”.



3. How to add more wheels?

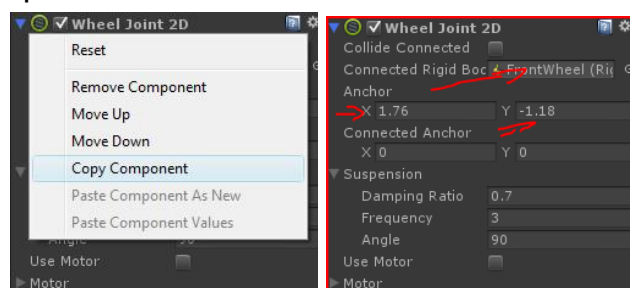
In the 2D physics mode you don't have wheel axis. if you want add more wheels do as follows:

1. Copy one of the current wheels (Ctrl+D) and rename it a you wish (eg. 'Third wheel')

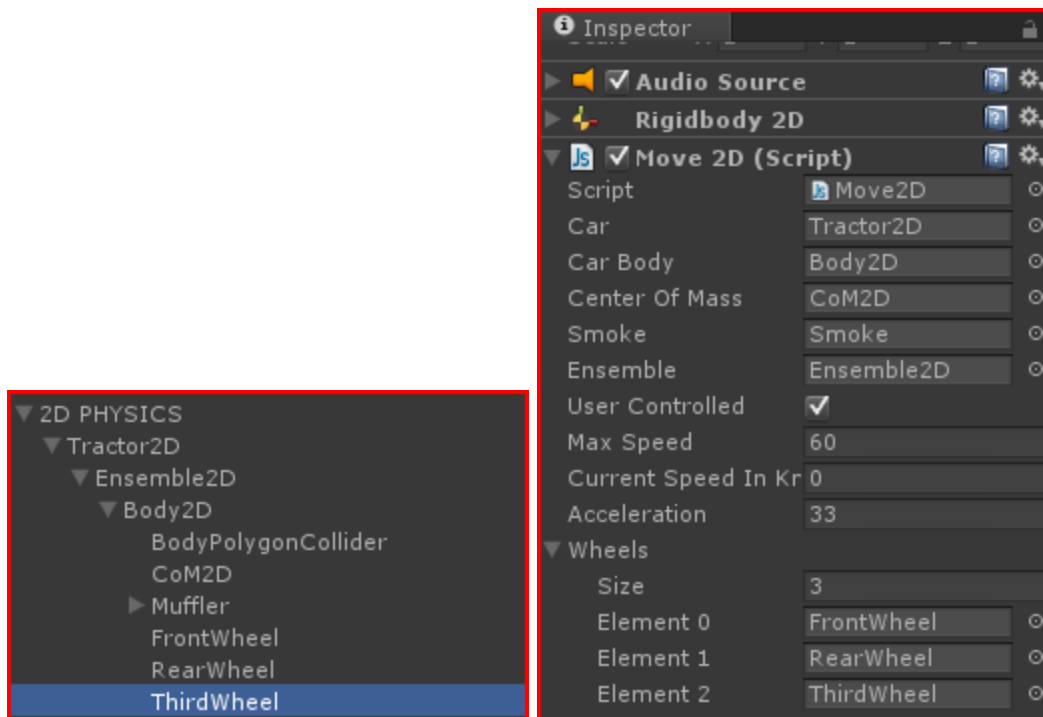


2. Select 'Body2D' object in the 'Hirarchy' window and then copy one of the existing 'Wheel Joint 2D' components and 'Paste as new component'

3. Drag 'Third wheel' object into 'Connected Rigid Body' field of the newly creted 'Wheel Joint 2D' component.



4. Select 'Tractor2D' object and then drag 'ThirdWheel' object to the 'Wheels' of the 'Move 2D' script component.



Contact

If you have any questions, please don't hesitate to contact me:

marcjan.b@gmail.com