VehicleSim Dynamics Plugin for Unreal Engine Tutorial

Before You Start.....................................................................................................1
How to Setup the VehicleSim Dynamics Plugin....................................................2
How to Attach Objects to the Simulation...............................................................4
Loading an Example Project ...............................................................................5
Creating a Vehicle Model in CarSim ......................................................................8
Changing CarSim Vehicle Parameters ..................................................................11
Selecting a Vehicle with a different type of Suspension ......................................12
TruckSim Only: Removing the Trailer .................................................................14
Input and Configuration Settings of the Plugin ....................................................15
Changing the Coefficient of Road Friction ..........................................................17
Using TruckSim in the plugin ..............................................................................19
Path Utility ...........................................................................................................20

Unreal Engine is a powerful gaming engine that provides high-quality graphics. The Mechanical Simulation VehicleSim Dynamics Plugin for Unreal Engine ("the plugin") allows users to create these high-quality graphics and include CarSim and TruckSim solvers. The plugin will feed terrain data, including surface friction, from Unreal Engine to the CarSim or TruckSim math model. It will also provide the math model data to the simulation using C++ code or Blueprints (Unreal Engine’s block-based visual scripting system). Standard CarSim output files are generated so that you can view the selected variable outputs in the VS Visualizer plotter (included with CarSim and TruckSim) after a run.

This tutorial assumes you have little or no experience with Unreal Engine and that you would like to use the plugin for your vehicle simulations.

This tutorial uses the CarSim model in the VehicleSim plugin for Unreal; TruckSim may be used with the plugin using similar steps.

Before You Start
Before starting, there are a few requirements.

1. CarSim 2019.1 must be installed on your computer.
2. You must have a valid CarSim 2019.1 license. Ensure that either the main CarSim program or the CLSM license utility is running with the CarSim Solver license feature enabled.
How to Setup the VehicleSim Dynamics Plugin

Once you’ve downloaded and installed Unreal Engine, open the Epic Game Launcher to access the application. Go to Marketplace, and search for “VehicleSim” to find the plugin (Figure 1). Click **Install to Engine** 1, then click the **Launch** button to access the project browser.

![Figure 1. VehicleSim Dynamics plugin installation.](image)

In the Unreal Project Browser (Figure 2), go to the tab labeled **New Project** 1, and from there, navigate to the **Blueprint** tab (you can create a C++ project if you prefer). Select **Vehicle** from the list of options 3, then click the **Create Project** button in the lower right corner 4.

**Note** The VehicleSim Dynamics plugin must be enabled through the Unreal browser before you have access to it. If you are not prompted by the engine to do so, you can navigate from the Unreal browser to **Edit > Plugins**, where you should click the **Enable** checkbox (Figure 3). You will then be prompted to restart Unreal; please follow the prompt and restart.

This creates a new project using the Blueprint **Vehicle Template Project**. Once you’ve created the project, navigate to the left side of the screen under **Modes** 1 (Figure 4) and search for CarSim 2. An item named **Carsim Vehicle Pawn** will show up under the search bar. You have now located the proper plugin component we will be using with the Unreal Plugin.
Figure 2. Unreal project browser.

Figure 3. Plugin Browser Screen.

Figure 4. Finding the CarSim Pawn.
How to Attach Objects to the Simulation

In the following steps, we will add a body object to the project, attach a body mesh to it, and finally adjust the offset coordinates such that the overlay matches the origin of the CarSim coordinate system.

1. Drag the **CarSim Vehicle Pawn** to the Unreal Viewport (Figure 5). We have now located the correct class object to use with the plugin and have dragged it into the window from the menu.

![Figure 5. Navigating the Unreal GUI.](image)

**Note** The vehicle will not have any body geometry attached initially. It will appear as an xyz origin with cameras, as shown above.

2. Toward the top right side of the Unreal Editor screen is a menu box called the **World Outliner**. Locate the **CarSimPawn** and highlight by left-clicking once.

3. At the lower part of the screen is the project space. Navigate to the example vehicle body in the content browser **Content > Vehicle > Sedan** and drag **Sedan_SkelMesh** to the **Details** menu, dropping the mesh onto **BodyShapeOffset**. It will prompt you for a name; press Enter to save as **Sedan_SkelMesh**. This will attach the body mesh to the **CarSimPawn** object.

4. Navigate to the **Sedan_SkelMesh** details (Figure 6). Here, you will find physics and collision elements to manage. Scroll down until you find the **Collision Presets**, and use the drop-down menu to change collision preset to **Vehicle** on the body mesh.
Figure 6. Set the collision presets to vehicle.

**Note**  Both the vehicle from the example project (Sedan) and the CarSim vehicle have their Auto Possess Player option set to Player 0, which may cause some confusion if you’re not aware of this naming convention. To avoid this, set Auto Possess Player to **Disable** for the Sedan.

Hit the **Play** button on the top of the Unreal Editor to drive the CarSim vehicle using the existing control mapping from the Sedan project (Figure 6) and click the **Viewport**. If a keyboard is used, the vehicle can be maneuvered with the WADS or arrow keys. Alternatively, you could also operate a gamepad or a gaming steering wheel kit to drive the vehicle.

After stopping the simulation (if a mouse cursor is not visible, hit “Escape”), the CarSim output files can be found in:

User home folder > AppData > Local > UnrealEngine > Common > CarSim_plugin

Note that the AppData folder is often treated as a Hidden Item in Windows Explorer. Click the checkbox to show **Hidden Items** under the **VIEW** tab of the explorer window.

**Loading an Example Project**

We have tested this example using CarSim 2019.1 and Unreal Engine version 4.22.

1. From the Unreal Marketplace, search CarSim. Click CarSim Vehicle Dynamics plug in icon (Figure 1. VehicleSim Dynamics plugin installation.). The link to the Example Project zip file is found at the bottom of the **Technical Details** section. Download the zip file and unzip into **Unreal Project** folder.
2. Launch Unreal Engine.

3. In the Unreal Project Browser, select the Example_UE4_Project.uproject file. If this example project file was not saved in the Unreal Project folder, browse and find the project file.

   **Note** If a message appears saying modules are missing or requesting to build with a different engine, click **Yes**. This occurs when a newer version of Unreal Engine is used.

4. The Unreal Editor will open the project with two vehicles in a driving environment (Figure 7).

   ![Figure 7. Example project.](image)

5. The VehicleSim Dynamics plugin requires a CarSim license to run. To enable the CarSim license, ensure that either:
   a. The main CarSim program (carsim.exe) is running, or
   b. The VLSM license utility is running with the CarSim Solver license feature enabled.
6. Hit the **Play** button on the top of Unreal Engine to drive the CarSim vehicle using the existing control mapping form the example project. The vehicle can be moved using a keyboard with WASD or arrow keys. A gamepad may also be used.

**Note**  
The Marketplace project contains two vehicles in order to support the VS Connect example. When testing a single vehicle as outlined in this document, you will receive multiple warnings indicating “Unable to locate VS Solver Variable.” These can be safely ignored. The second vehicle, named `A_MovingObjectVehicle` in the **World Outliner**, will set itself to invisible when **Play** is hit if it is not being updated via VS Connect. For more details about using VS Connect with The CarSim Unreal Plugin, see:  
http://www.unrealengine.com/marketplace/carsim-vehicle-dynamics

If a user-specified CarSim model is not provided, Unreal Engine uses a default CarSim model. The default CarSim model is included in the VehicleSim Dynamics Plugin directory, and it is found in:

Unreal Engine plugin directory: Plugins > Marketplace > CarSim > ThirdParty > CarSim

**Note**  
When you hit the Play button, and the animation looks like it in 3D mode, you need to turn off Stereoscopic 3D Mode in NVIDIA control panel (Figure 8).

*Figure 8. Disabling Stereoscopic 3D Mode in NVIDIA control panel.*
Creating a Vehicle Model in CarSim

As mentioned in the previous section, Unreal Engine uses a default VehicleSim Dynamics Plugin model if a vehicle model is not specified. This section describes how to set a CarSim model and use it to run inside Unreal Engine.

Open CarSim, and use the menu command File > New Dataset (Empty) to create a new dataset in the Run Control library. Set the Category for the new dataset to Unreal Engine Plug-in (1), and in Title for the new dataset, enter UE Vehicle (2) (Figure 9).

Figure 9. Creating a new CarSim dataset.

In this new dataset (Figure 10), uncheck Set driver controls here (1), Set time step here (2), and Advanced settings (3). Use the pull-down menu next to Math Model and select Vehicles >
Vehicle Assembly. Then, select D-Class, Sedan w/ 5 Motion Sensors. Select 0 from the More plots drop down menu. Select Vehicle Reference > 13 Azm, 9 El, 18 m Dist, 45 FoV, No Yaw for Animator: Camera Setup dataset.

Click D-Class, Sedan blue link and go to Brake System dataset. Change the brake control system to Control with Pedal Force (Figure 11).

![Image of Brakes: Four-Wheel System]

*Figure 11. Select Control with Pedal Force.*

**Note** Currently, the braking can only be controlled by the pedal force using Unreal Engine plug-in. You need to select Control with pedal force option in the brake dataset.

Next, we will create another dataset which will be used to generate the simfile to connect the vehicle dynamics to the plugin. After setting up the vehicle model, select Models: Transfer to Local Windows Directory from the Models drop-down menu. Select [Link to New Dataset] (Figure 12). In Title for the new dataset, enter UE4 simfile generator.

![Image of Simulated Test Specifications, Run from Alternate Directory, and Analyze Results (Post Processing)]

*Figure 12. Select Models: Transfer to Local Windows Directory and select [Link to New Dataset]*

Click the blue link to enter to this newly created UE4 simfile generator dataset (Note: This UE4 simfile generator dataset is provided in CarSim 2019.1).

**UE4 simfile generator** dataset should contain the information shown in Figure 13. The working directory should contain a period, “.”. When there is a period, “.”, in this field, CarSim will create a simfile in the current database directory. Select to use 64-bit solver. Write simfile
ue4simfile.sim in the yellow field ①. In this example, the simfile name is specified as ue4simfile.sim ③.

Figure 13. UE4 simfile generator dataset Models screen.

Next, go back to the Run Control screen, and click Generate Files for This Run button. Then, navigate to the working directory you just set up (the current database), locate the file ue4simfile.sim that you just created, and press Shift + Right-Click on it. Select Copy as Path from the resulting dropdown-menu.

Go back to the Unreal Editor. Find MSC_DemoVehicle in World Outliner and right-click to highlight it. Then, in the Details tab, scroll down to find and click CarSimMovement (Inherited) to highlight the component ① (Figure 14).

Figure 14. CarSim movement using Unreal.
In the Unreal Editor, still highlighting CarSimMovement (Inherited) component, scroll down to find CarSim > Solver Config File and Paste the path you copied in the previous step (Figure 15).

| Note | The path created using Copy Path in Windows Explorer includes a quotation mark, "", at the beginning and at the end of the path. Delete these two quotation marks in the pathname. |

Press the Play button in the Unreal Editor and then click the ViewPort. CarSim will now run with the given parameters. The vehicle can be moved using a keyboard with WASD or arrow keys. A gamepad is also used. Press H key to display a speedometer and a tachometer. Press the Tab key to toggle the camera view between the driver view and the outside view. After stopping the simulation (if a mouse cursor is not visible, hit “Escape”), the CarSim output files can be found from the CarSim GUI. Go to View > Open Results Folder for this Run in Windows.

![Carsim](image)

*Figure 15. Vs Config File Setup.*

### Changing CarSim Vehicle Parameters

You can now modify the vehicle inside CarSim and test it using the Unreal Editor.

1. From the Run Control screen in CarSim, click the arrow next to the vehicle dataset and select [Copy and Link Dataset] to duplicate it. Leave the category name and the title name as the default (Figure 16).

2. Click the blue link D-Class, Sedan w/ 5 Motion Sensors #1, select [Copy and Link Dataset] to duplicate the Lead Unit: Ind_Ind dataset.
   a. Name the title D-Class, Sedan 2019.1 #1 and click the link.
b. Repeat the same thing and duplicate the **Rigid Sprung Mass** dataset, giving it the new title **D-Class, Sedan #1**.

![Simulated Test Specifications](image)

*Figure 16. Duplicate Dataset in CarSim.*

3. Navigate to the sprung mass dataset by clicking the blue link again, and change the sprung mass value from 1370 Kg (Figure 17) to 1600 Kg.

4. Click the **Back** button to get back to the **Vehicle: Assembly** screen. Change the tires to **Touring Tires: 265/70 R17**.

5. Click the **Home** button in CarSim to get back to the Run Control Screen, and then click **Generate Files for this Run** to update ue4simfile.sim.

6. Go back to the Unreal Editor, and press the **Play** button again. This time the vehicle runs with the updated vehicle model with the new sprung mass. Note that since you have not changed the skin in UE4, the vehicle animation in UE4 looks same as before.

The CarSim output files can be found in Results/ folder.

**Selecting a Vehicle with a different type of Suspension**

It is possible to run a different suspension vehicle using the above example. The instruction in this section is also applied for the front/rear independent suspension vehicle.
1. Go to CarSim Run Control, and change the vehicle type by selecting **Vehicles > Vehicle with Loads, Sensors, Trailer, etc.**, and then select from that list **Pickup, Full Size w/ 2000kg Payload**, which has a rear solid axle suspension (Ind_SA) (Figure 18).

   **Note** Braking can only be controlled by the pedal force when using Unreal Engine plug-in. You need to select Control with pedal force option in the brake dataset.

2. Press the **Generate Files for this Run** button. This will update ue4simfile.sim.

3. Go back to the Unreal Editor, and press the **Play** button again and click the **Viewport**. The vehicle will now run with the updated vehicle model. Note that the vehicle animation in UE4 looks same unless you update the skin in UE4.

The results of this run can be found in your CarSim Database results directory. In CarSim, navigate to **View > Open Results Folder for this Run in Windows** to find the results.

---

**Figure 17. Changing the vehicle sprung mass in CarSim.**
Figure 18. Selecting a different vehicle and suspension.

**TruckSim Only: Removing the Trailer**

1. TruckSim includes *Unreal Engine 4 Plug-in, Lead Vehicle* example. Using this example and by following previous sections, it is easy to run TruckSim inside Unreal Engine environment. To remove the trailer, go to this example, click **UE4 Default** blue link under *Math Model: S_S + S* and go inside “**Vehicle: Loaded Combination; UE4 Default**” dataset. Uncheck Trailer (Figure 19).
2. Hit the Home button to go back to Run Control screen. Verify that it is updated to **Math Model: S_S**.

3. Hit the **Generate Files for this Run** button to create a new simfile.

Now, while the truck is running, there is no trailer attached. This is because the trailer shape is included in the project and it is added and removed dynamically inside VehicleSim Dynamic plugin.

**Input and Configuration Settings of the Plugin**

You can adjust the keyboard settings using Settings option in the Unreal Editor.

Go to **Settings > Projects Settings**. Select **Input** under **Engine** category (Figure 20). Here, you can set the different keyboard control and the scale of the throttle and the brake. In Figure 20, it is showing that the Brake is 1.0 when using “S” and 0.5 but when using “Down” key. That means, the brake force is applied 50% of the force when using “Down” key.

You can select different **CarSim** settings in the **Details** tab as follows (Figure 21).

The **Solver Config File** is where you can enter the name of CarSim *.sim* (Simfile). To use the example models packaged with the plugin, you can simply enter **carsim_all.par** or **trucksim_all.par** depending on which product you’re using. If you’d like to inspect these files, they are in your plugin installation location from the Unreal Marketplace. The default location would be **Engine\Plugins\Marketplace\CarSim\ThirdParty\VehicleSim** inside of your engine installation, e.g. **c:\Program Files\Epic Games\UE_4.22**.
Instead of the road in the Unreal scene, you can use the CarSim Road. To do so, select **Use CarSim Road**. If this box is checked, the CarSim solver uses the road defined in CarSim. If this box is not checked, the solver will query the Unreal terrain below the vehicle tires. If **Use VehicleSim start position** is checked, the vehicle will be positioned based on the parameters in the Parsfile.

![Figure 20. Unreal Engine Input Settings.](image-url)
Figure 21. CarSim configuration settings.

For the driver control, Use CarSim Driver box can be checked/unchecked. If the box is checked, CarSim driver model is used and the control is ignored. If unchecked, the vehicle inputs will be passed to the solver but the CarSim driver model will be suppressed. Note that it is important to match the CarSim coordinates and the UE4 coordinates when use CarSim driver model in UE4. For example, if your vehicle’s starting position is at (500, 500, 0) in UE4 but the CarSim starts at (0, 0, 0), you are not able to observe a desired maneuver.

If you’d like to keep a vehicle in your scene but not have it activated you can select Disable VehicleSim vehicle. This will prevent the vehicle from doing a license check and loading when the simulation begins. This can be useful for having many vehicles in your scene that you can activate as needed. The example scene will check this flag during Begin Play in the blueprints and hide the vehicle from the scene if it is disabled.

If the driver control box is unchecked, Max Hand Wheel Angle Degrees, Max Throttle, and Max Brake Pedal Force Newtons can be adjusted.

Changing the Coefficient of Road Friction
If you are not using CarSim Road (in Figure 21), and if you want to change the road friction coefficient, you can do so inside Unreal Editor.

To change the road friction for the Example_UP4_Project.uproject, in Content Browser, go to Content > Environment > Materials folder (Figure 22).
Figure 22. Access to the Materials from Content Browser.

In the right bottom corner of the Content Browser tab, click View Options, check Column for the View Type, and sort with Type so it is easy to find Physical Materials (Figure 23).

In this example, the gray part of the road uses Road PhysMat physical material. Click road_PhysMat. When the window opens, the Friction entry is found under Physical Material (Figure 24). Change this value to experiment.

The white part of the road uses Concrete_PhysMat material. If necessary, change this friction coefficient as well.

Figure 23. Select Column for the View Options.
Using TruckSim in the plugin

New for the 2019.1 release, you can use TruckSim solvers with the plugin. The setup and configuration are the same as outlined above for CarSim, although the blueprints are different to accommodate trailer spawning and uses appropriate shapes for the default truck model. As with the car, you can customize these to your needs or create your own blueprints.

The example project is by default setup to run CarSim. The Disable VehicleSim vehicle checkbox (see Figure 15) is selected for the truck model, which will not allow the vehicle to initialize. The truck blueprint is configured to hide it from the scene when disabled.

If you want to use the truck you will need to turn off the Disable VehicleSim vehicle option and set the pawn to Auto Possess Player with your active player (Figure 25), usually Player 0. You may also want to disable and hide the car if you’re using the truck.
Figure 25. Setting the TruckSim pawn to be possessed by Player 0.

Path Utility
The Example Project includes Path Utility blueprint. Using this utility, the coordinates of the path can be easily obtained from Unreal Engine and imported to CarSim. To use this utility, follow the steps below:

1. In Content Browser, go to Content/VS_Core and find PathUtility (Figure 27). Grab this PathUtility icon and drag it to the ViewPort. Use a mouse and move it to the location of the desired path starting point.

Figure 26. PathUtility in Content Browser.
2. Once PathUtility is added in ViewPort, it can be found in World Outliner (Figure 27).

![Figure 27. PathUtility in World outliner.](image)

3. Select PathUtility in World Outliner and select PathUtility (self) in Details (Figure 28). To add points, scroll down and go to Default and click Add Spline Point. Each time a spline point is added, move the point to the desired location by using a mouse. Make sure z-height of the path is above ground. A point can be deleted by clicking and highlighting the point and press the delete button. Add the desired numbers of points for the path in Points Along Path. The default is set to 100.
4. To write x, y, z coordinates of the path, set the name of a csv file in **Path File**, and press the **Print Path Spline to File** button.

5. Open the csv file. Copy and paste the x, y, z coordinates of the path into CarSim (Figure 29).

6. After entering the new path in VehicleSim, generate a new simfile.sim and use this simfile as **solver config file**. Make sure **Use VehicleSim start position** and **Use VehicleSim driver** are selected.
Note that the users do not need to worry about the z-height of the path as long as Use VehicleSim road is unchecked. When this option is unchecked, the solver will query the Unreal Engine terrain below the tires.

**IMPORTANT** The z-height of the path does not have to be accurate if Use VehicleSim road option is unchecked. However, when defining a path (in Step 3 above), make sure that z-height is above ground. Otherwise, the solver cannot find the terrain information under the tires.