AG RACING KIT

Physically Based Anti Gravity Racing

AG Racing Kit Documentation 2.0.X

Getting Started - Follow these steps if you are here the first time!

Setting up the Game Mode

Setting up Input Conrols

AG Vehicle

Entering the Vehicle

Vehicle Setup

Spline Track

Creating a Spline Track

Adding Spline Objects

Racing Manager

Purposes of the Racing Manager

Racing Manager Setup

Different Race Types

Setting up the Game Mode

To make the system compatible with your project, you first have to set up the Game Mode Blueprint.

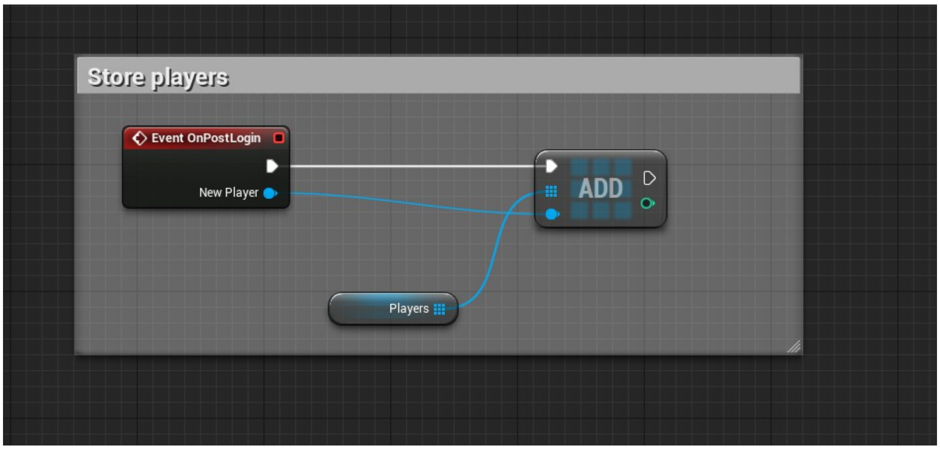
There are two ways to achieve this:

Option A: Using the default Game Mode which is included in the package

- Go to **Edit -> Project Settings** in the Engine Menu Bar
- Select **Maps & Modes** under Projects in the left side of the Project Settings window
- At the Default GameMode dropdown, set **AG_Gamemode** as your default Game Mode

Option B: Create new or use existing Game Mode Blueprint

To make your own Game Mode Blueprint compatible with the Racing Manager, all you have to do is to store new players in a Player Controller Array and set it as your default Game Mode.



Setting up Input Mappings

For the vehicle to be able to receive input controls, the Axis Mappings have to be set up properly. Again, there are two ways to achieve this. You can either set the input mappings manually or use the input.ini file.

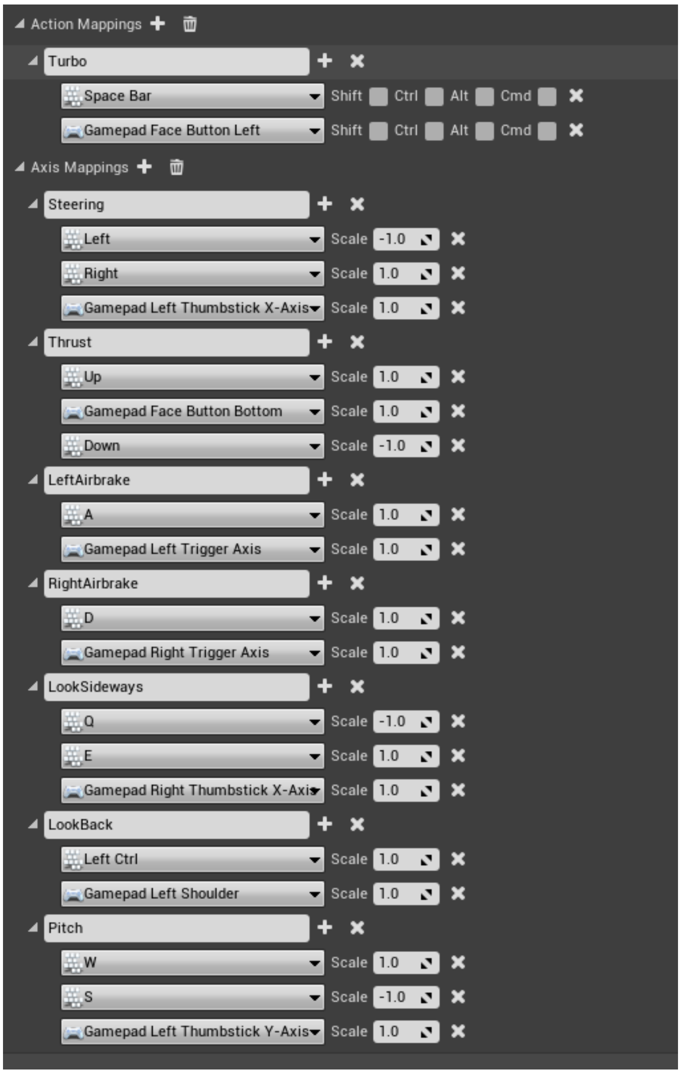
If you already have input mappings set up for your project, I recommend you to set them up manually since importing the input file will override all existing input mappings!

Download and import the Input file

- Download the Input File from the Website
- Go to **Edit -> Project Settings** in the Engine Menu Bar
- Under **Engine -> Input** select the 'Import' Button at the top right and import the .ini file

Create input mappings manually

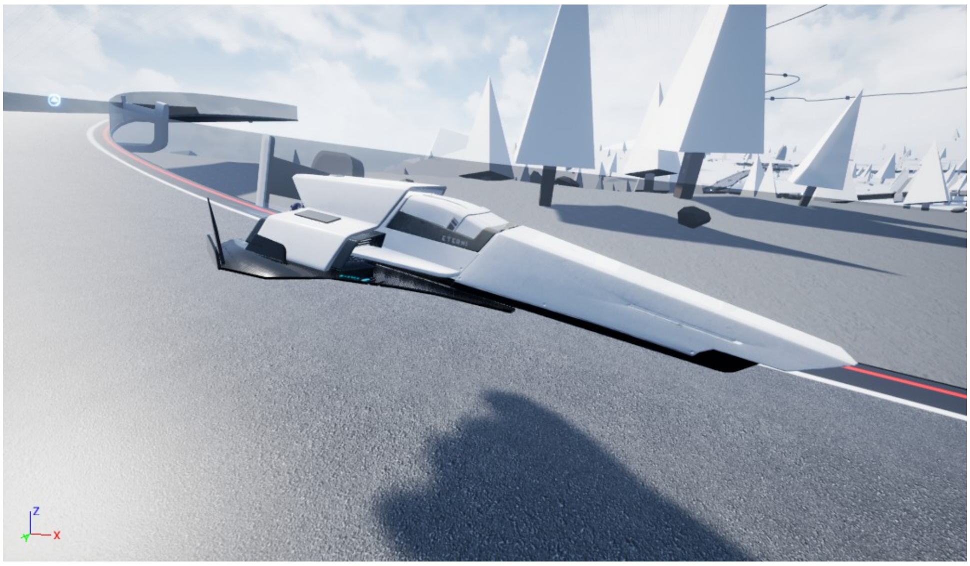
To set up input mappings correctly, apply these input mappings from the following screenshot. The input controls are case sensitive, so check twice that they are written in the correct way.



Action Mappings
- Turbo
  - Space Bar — Shift □ Ctrl □ Alt □ Cmd □
  - Gamepad Face Button Left — Shift □ Ctrl □ Alt □ Cmd □

Axis Mappings
- Steering
  - Left — Scale -1.0
  - Right — Scale 1.0
  - Gamepad Left Thumbstick X-Axis — Scale 1.0
- Thrust
  - Up — Scale 1.0
  - Gamepad Face Button Bottom — Scale 1.0
  - Down — Scale -1.0
- LeftAirbrake
  - A — Scale 1.0
  - Gamepad Left Trigger Axis — Scale 1.0
- RightAirbrake
  - D — Scale 1.0
  - Gamepad Right Trigger Axis — Scale 1.0
- LookSideways
  - Q — Scale -1.0
  - E — Scale 1.0
  - Gamepad Right Thumbstick X-Axis — Scale 1.0
- LookBack
  - Left Ctrl — Scale 1.0
  - Gamepad Left Shoulder — Scale 1.0
- Pitch
  - W — Scale 1.0
  - S — Scale -1.0
  - Gamepad Left Thumbstick Y-Axis — Scale 1.0

AG Vehicle

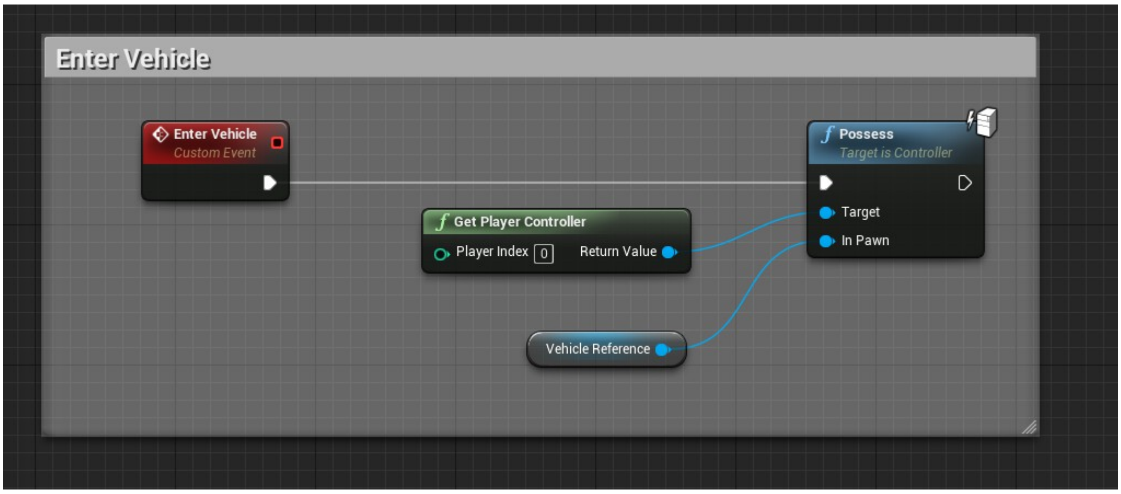


Entering the vehicle

To use the vehicle as a standalone Actor, you can simply do this by possessing the Player Controller to the vehicle Pawn.

Let's say you have a Third Person Controller and just want to enter the vehicle as an addition to your game mechanics.

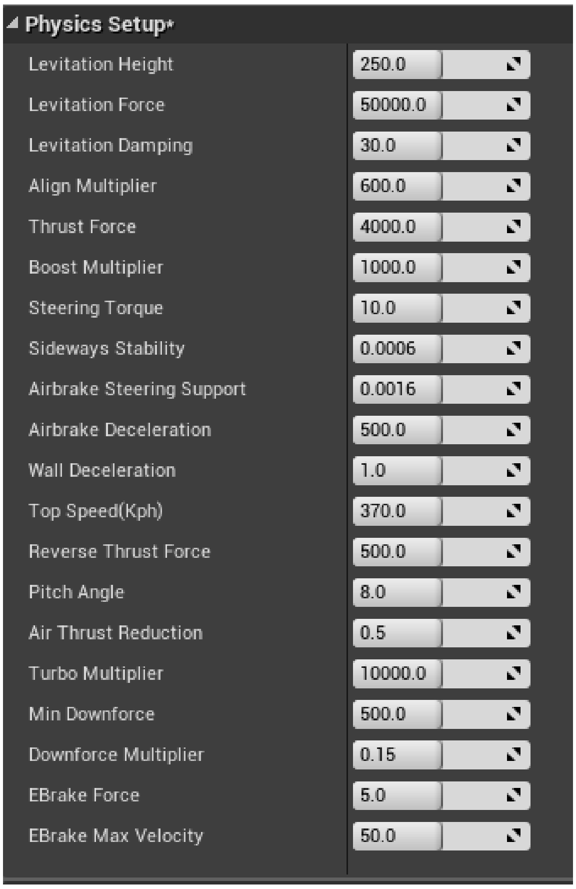
1. First make sure all input mappings are set up correctly for the vehicle to be able to receive input controls
2. Place or spawn the vehicle into your level and enter the vehicle Pawn using the *Possess* node



It doesn't matter from which Blueprint you call this node. The Racing Manager Blueprint also uses this node to possess the Players to the target vehicles.

Vehicle Setup

Physics Setup



Physics Setup*

Levitation Height	250.0
Levitation Force	50000.0
Levitation Damping	30.0
Align Multiplier	600.0
Thrust Force	4000.0
Boost Multiplier	1000.0
Steering Torque	10.0
Sideways Stability	0.0006
Airbrake Steering Support	0.0016
Airbrake Deceleration	500.0
Wall Deceleration	1.0
Top Speed(Kph)	370.0
Reverse Thrust Force	500.0
Pitch Angle	8.0
Air Thrust Reduction	0.5
Turbo Multiplier	10000.0
Min Downforce	500.0
Downforce Multiplier	0.15
EBrake Force	5.0
EBrake Max Velocity	50.0

Levitation Height
Target distance between the surface and the vehicle

Levitation Force
Maximum repulsion force against the surface

Levitation Damping
Damping rate or 'smoothness' of the hover motion

Align Multiplier
Speed of the Vehicle alignment to the surface

Thrust Force
Default forward acceleration force

Boost Multiplier
Additional acceleration force the Vehicle gets from a triggered Boostpad

Sideways Stability
Stability of the vehicle when turning - set this low for a drift setup or set it high for a F1 style setup

Airbrake Steering Support
Additional steering torque when using the Airbrakes

Airbrake Deceleration
Deceleration force of each Airbrake

Wall Deceleration
Deceleration force the vehicle experiences when it has contact to a side wall (must have the WallCollision Physic Material assigned)

Top Speed
Maximum speed to which the vehicle can accelerate. The Thrust force decreases linearly to this specific velocity where it stops accelerating. This feature replaces the previous drag value where you had to balance thrust force / drag to find the perfect top speed for your needs

Reverse Thrust Force
Reverse acceleration force when the Vehicle drives backwards

Pitch Angle
Angle up to which the vehicle can manage it's relative pitch to the surface

Air Thrust Reduction
Multiplier of the Thrust Force when the vehicle is in the air. This feature prevents the Vehicle from taking off when the Line Traces have no contact to the ground.

Turbo Multiplier
Additional boost force the vehicle gets when using the Turbo function

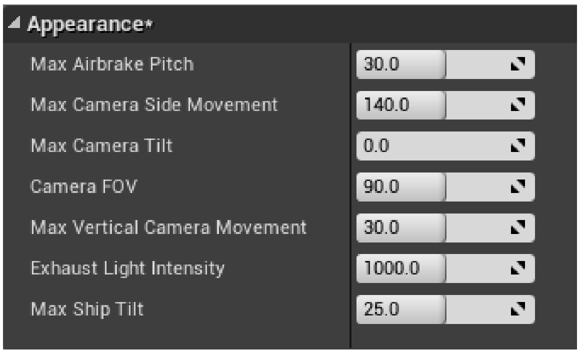
Min Downforce
Clamps the downforce of the vehicle to a minimum - can be used for additional gravity without changing the gravity in the world settings

Downforce Multiplier
Rate of which the Vehicle gets pressed onto the ground the faster it gets

EBrake Force
Force of the E-Brake system to hold the vehicle on one place

EBrake Max Velocity
Maximum velocity to which the EBrake system can operate

Appearance



Appearance*

Max Airbrake Pitch	30.0
Max Camera Side Movement	140.0
Max Camera Tilt	0.0
Camera FOV	90.0
Max Vertical Camera Movement	30.0
Exhaust Light Intensity	1000.0
Max Ship Tilt	25.0

Max Airbrake Pitch
Maximum rotation (in degrees) to which the Airbrakes can rotate

Max Camera Side Movement
Maximum relative sideways movement of the camera while steering

Max Camera Tilt
Maximum tilt (in degrees) to which the Airbrakes can rotate

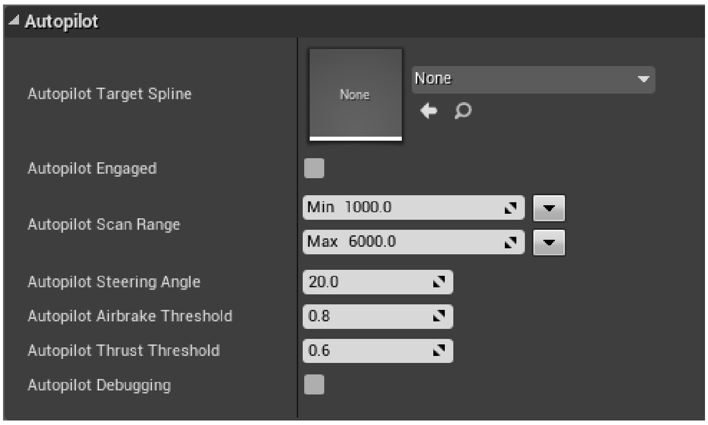
Camera FOV
Default Field of View of the player camera

Max Vertical Camera Movement
Maximum relative vertical movement of the camera

Exhaust Light Intensity
Light intensity of the illuminated exhaust area

Max Ship Tilt
Maximum tilt of the vehicle mesh (in degrees) while steering

Autopilot



Autopilot

Autopilot Target Spline: None

Autopilot Engaged: □

Autopilot Scan Range: Min 1000.0, Max 6000.0

Autopilot Steering Angle: 20.0

Autopilot Airbrake Threshold: 0.8

Autopilot Thrust Threshold: 0.6

Autopilot Debugging: □

Autopilot Target Spline
Target Spline Component for the Autopilot to follow

Autopilot Engaged
Activates / Deactivates Autopilot controls

Autopilot Scan Range
Min and Max distance between the Ship and the Autopilot scan point along the Spline

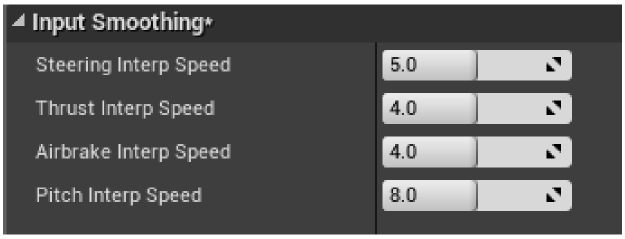
Autopilot Steering Angle
Max angle between the Ship and the Autopilot scan point

Autopilot Airbrake Threshold
Minimum absolute steering value (between 0-1) where the Airbrakes gets activated for better cornering

Autopilot Thrust Threshold
Minimum absolute steering value (between 0-1) where the throttle gets released to prevent the Vehicle from driving into curves too fast

Autopilot Debugging
Shows where the scan point is along the spline and draws the distance between the ship and the point

Input Smoothing



Input Smoothing*

Steering Interp Speed: 5.0

Thrust Interp Speed: 4.0

Airbrake Interp Speed: 4.0

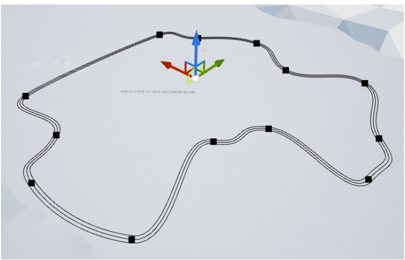
Pitch Interp Speed: 8.0

Interp Speed Values

Input smoothing uses linear interpolation to create smooth input controls for a more realistic driving experience. These values determine the speed of the linear interpolation. Setting them high will cause the Vehicle to respond more directly to input controls.

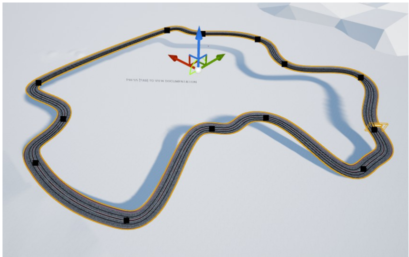
Spline Track

Creating a Spline Track



1. Drag the Blueprint into your level

When you drag the Spline Track Blueprint into your level, the procedural track generator automatically creates a spline shape. When you take a look at the Details Panel at the *Procedural Track* section, you can tweak the values until you are satisfied with the shape.

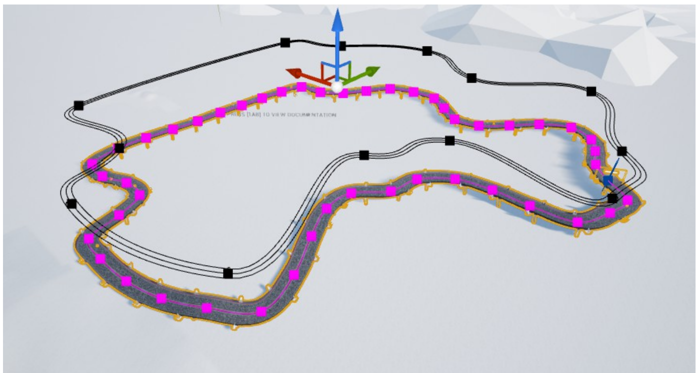


2. Assign a Track Mesh

In the Details Panel, go to the *Spline Tracks* section and add a Track to the *Tracks* array by clicking on the + button. Select a Track Mesh (in this case the sample track) and you have a complete Spline Track. Depending on your mesh, you can also refine the *Resolution* of your track mesh.

3. Align to Terrain

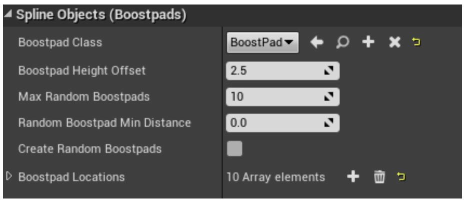
If you want to have your track aligned to the terrain underneath, all you have to do is check the *Align To Terrain* option in the *Track Properties*. You can elevate the track with the *Terrain Offset* value to prevent it from intersecting with the terrain. With the Target Segment Length value you can set the overall resolution of the Spline that gets projected onto the terrain.



Adding Spline Objects

Boostpads

To create Boostpads along the Spline Track, you can either add Boostpads manually by adding elements to the *Boostpad Locations Array,* or click on the *Create Random Boostpads* to place them procedurally through the Blueprint.

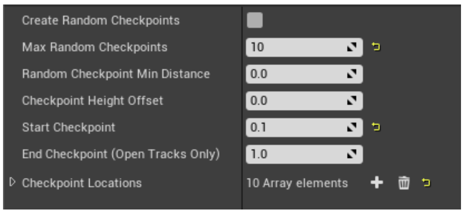


Spline Objects (Boostpads)

Boostpad Class	BoostPad
Boostpad Height Offset	2.5
Max Random Boostpads	10
Random Boostpad Min Distance	0.0
Create Random Boostpads	□
Boostpad Locations	10 Array elements

Checkpoints

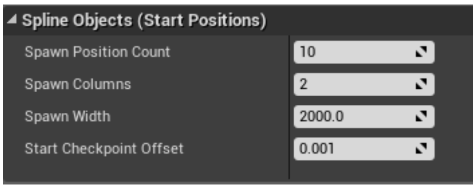
Similar to the Boostpads, you can add Checkpoints procedurally by clicking the *Create Random Checkpoints* checkbox. Random Checkpoint Min Distance determines the minimum distance in track units (one track length = 1) between the procedurally placed Checkpoints.



Create Random Checkpoints	□
Max Random Checkpoints	10
Random Checkpoint Min Distance	0.0
Checkpoint Height Offset	0.0
Start Checkpoint	0.1
End Checkpoint (Open Tracks Only)	1.0
Checkpoint Locations	10 Array elements

Start Positions

Player Starts can also be placed through the Spline Blueprint. To create Player Starts, first set the number of spawn points with the *Spawn Position Count* value. Spawn Columns and Spawn Width tell the system how wide the columns are spread from each other on the track.

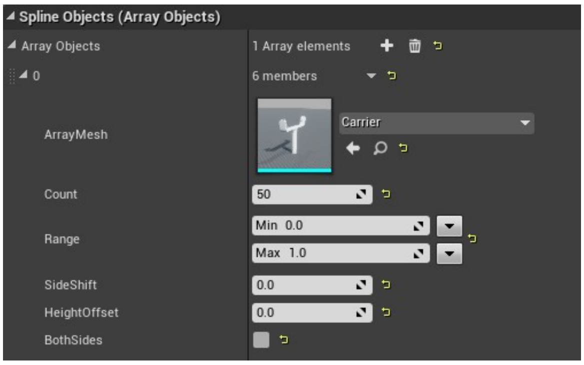


Spline Objects (Start Positions)

Spawn Position Count	10
Spawn Columns	2
Spawn Width	2000.0
Start Checkpoint Offset	0.001

Array Meshes

With Array Objects you can add meshes that repeat themselves frequently (for example carriers, lights, poles etc.). Simply Add an element with the + button to *Array Objects*. With the *Side Shift* value, you can create an offset so that the objects are either left or right relative to the track. If you want meshes on both sides, check the *Both Sides* option.



Spline Objects (Array Objects)

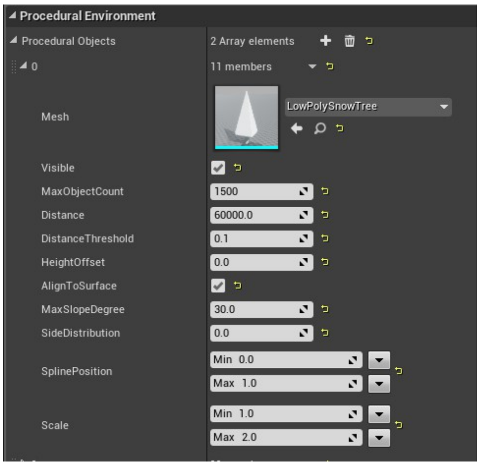
Array Objects	1 Array elements
0	6 members
ArrayMesh	Carrier
Count	50
Range	Min 0.0 Max 1.0
SideShift	0.0
HeightOffset	0.0
BothSides	□

Custom Child Actors

For Actors other than Boostpads, Checkpoints, Player Starts or Array Meshes, you can use the *Custom Child Actors* Array to create Acotrs of any Class along the Spline Track.

Procedural Environment

The Spline Track Blueprint has the ability to spawn entire environments along the track. For each procedural object you can specify object count, track distance, spline range, scale and rotation. On one hand this is practical for prototyping, to create appealing environments with just a few objects. On the other hand, this is very useful for objects near the track, as they will move dynamically with the spline as you modify it – making sure no object will be placed on the track directly. Adding procedural objects works similar to the previous methods. You can add elements to the *Procedural Objects* Array and specify different settings for each object.



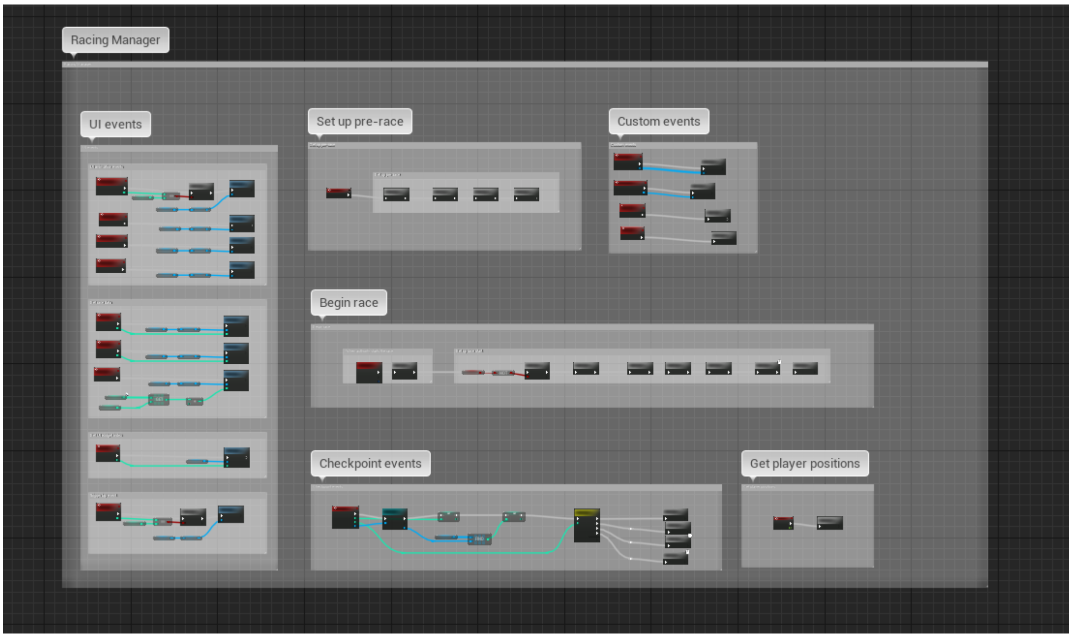
Procedural Environment

Procedural Objects	2 Array elements
0	11 members
Mesh	LowPolySnowTree
Visible	☑
MaxObjectCount	1500
Distance	60000.0
DistanceThreshold	0.1
HeightOffset	0.0
AlignToSurface	☑
MaxSlopeDegree	30.0
SideDistribution	0.0
SplinePosition	Min 0.0 Max 1.0
Scale	Min 1.0 Max 2.0

Racing Manager

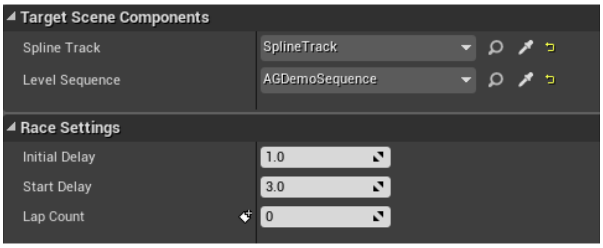
Purposes of the Racing Manager

The Racing Manager takes care of all the Race Logic behind the scenes. It is responsible for the race setup, player spawns, real time player positions, UI, and all checkpoint trigger events.



Setup

Setting up the racing manager is easy. All you have to do is to place the Racing Manager Blueprint anywhere in your level. It works hand in hand with the Spline Track Blueprint, so all you have to do is to assign the spline Track. Optionally, you can specify a Level Sequence which gets played as an intro before the race starts.



Target Scene Components

Spline Track	SplineTrack
Level Sequence	AGDemoSequence

Race Settings

Initial Delay	1.0
Start Delay	3.0
Lap Count	0

Race Types

The Racing Manager currently supports **3 different Race Types:**

Infinite Mode / Best Lap (closed Tracks)
Condition: Lap count = 0
This is the simplest of the Race Types. Infinite mode consists of a race start and an infinite amount of laps to be completed until someone decides to quit. The UI Lap Time Recorder is still active and displays the current and best lap times.

Circuit Race (closed Tracks)
Condition: Lap count > 0
A Circuit Race is a conventional race with x amount of laps to be completed. In contrast to infinite mode, this Race Type finishes when one player completes all the given amount of laps.

Sprint Race (open Tracks)
Condition: Spline Track -> Loop = False
The Sprint Race type automatically applies when the target Spline Track is not a closed loop. The race is finished when one player triggers the Finish Checkpoint.