Realistic VFX Pack

Contact:
saitamastudiobusiness@gmail.com

MANUAL

Saitama Studio

Table of contents

First Steps; **Must Read**

Below follows the installation process. For a correct and easy setup you should read this.

Built-in Pipeline (NOT AVAILABLE)

Not currently available. This pack focuses on the Unity Visual Effect Graph pack which is unavailable in the Built-in Pipeline.

URP (Universal Render Pipeline)

To get everything to work properly you need to follow some few basic steps.

1. Installing two Packages from the Unity Package Manager.
    a. Install **Post Processing** package from the Package Manager.
    b. Install **Visual Effect Graph** package from the Package Manager.
2. [Optional] In Edit/Project Settings/Graphics, enable HDR in all checkboxes.

Everything should work after following those two steps. If you open any of the Demo Scenes, it should look similar to the trailer and media images.

Below follows how to replicate the same setup in another scene:

1. In your Camera, make sure HDR **and** Post Processing are turned **On**.
2. Add a Volume component anywhere in your scene.
    a. Add the included Realistic VFX Pack Profile to the Volume component's Profile property.
    b. Set the Mode property to Global for it to show in the entire scene.
3. Alternatively to (2.) you can drag-and-drop the Volume prefab under Realistic VFX Pack/URP/Demo/Demo Source into your scene for the same effect.

HDRP (High Definition Render Pipeline)

To get everything to work properly you need to follow some few basic steps.

1. **[VERY IMPORTANT]** Remove the URP folder from the package. The Realistic VFX Pack folder should ONLY contain this README.pdf and ImportHDRP.unitypackage
2. Installing two Packages from the Unity Package Manager.
    a. Install **Post Processing** package from the Package Manager.
    b. Install **Visual Effect Graph** package from the Package Manager.
3. [Optional] In Edit/Project Settings/Graphics, enable HDR in all checkboxes.
4. Now it is safe to import the actual HDRP asset pack. Import the "ImportHDRP_Destroy_URP_First" Unity Package included in the asset root folder.

Everything should work after following those two steps. The asset works just as well in HDRP as in URP — it has complete feature parity and all assets will look good with correct settings — but the Demo scenes were not designed to look good in this pipeline.

Below follows how to replicate the same setup in another scene:

1. In your Camera, make sure HDR **and** Post Processing are turned **On**. This will probably already be correctly configured with a normal HDRP setup.
2. Add a Volume component anywhere in your scene.
    a. Add the included Realistic VFX Pack Profile to the Volume component's Profile property.
    b. Set the Mode property to Global for it to show in the entire scene.
3. Alternatively to (2.) you can drag-and-drop the Volume prefab under Realistic VFX Pack/HDRP/Demo/Demo Source into your scene for the same effect.

Creating a Path; Curve (**Should Read**)

Here we will show you how to set-up a path that any effect of your choice should move in.

Creating a path [read]

First of all, we need to create the curve.

1. Go to Realistic VFX Pack/Prefabs.
2. Here we find the prefab "Curve". Simply drag-and-drop the prefab into your scene.
   a. You will notice that the prefab has two children; "VFX (add vfx as child of me)" and "Points [DO NOT TOUCH]". As the description entails, you should **never need to**(*) touch the children of "Points [DO NOT TOUCH]". The other child "VFX" is explained further [here](#).
3. If you select the root "Curve" prefab (the object with the script 'Curve' attached), you will see a path that you can edit.

Editing the path

The path is a simple curve between some points. Here we will explain how you can edit the curve to move however you want.

1. The curve is made up of a:
   a. Blue curvy line
      i. This is the main curve that the VFX will follow.
   b. Green to red straight line
      i. *This shows how much the in-between points deviate in length from the average distance between two neighboring points.* In other words, to obtain smooth movement, **keep these lines as GREEN as possible**.
      ii. If the line between two points gets close to being red, consider adding a new point between the two points.
   c. White spheres (points)
      i. These are the points that make up the path. You can move these points by simply click-and-drag the handle.
2. Adding and removing points:

    a. To add or remove a new point, you navigate to the 'Curve' script under the "Curve" prefab. Here you can see two buttons. "NEW POINT"; creates a new point as the last point of the path. "REMOVE LAST POINT"; removes the last point of the path.

3. Additional settings:
    a. In the 'Curve' script, two more properties relating to the curve, "Closed"; whether the curve should connect the first and last points to create one continuous curve or not. "Curviness"; how 'curvy' the curve should be, values closer to 0 creates a more rigid path, values closer to 1 creates a more 'curvy' curve. The closer to **0.5**, the smoother the curve.

4. VFX related settings; each curve has some properties that change how the VFX should follow the path. These settings are applied to every VFX that are children of "VFX (add vfx as child of me)". Every VFX also has some multipliers to these properties for further individual VFX settings.
    a. "Radius"; offset the VFX particles can move away from the curve (thickness of curve).
    b. "Speed"; how fast the particles should move around the curve.
    c. "Follow Path Force"; the force to be applied to VFX particles to keep the particles on the path. Try to experiment with **both** "Speed" and "Follow Path Force" to change how fast the particles move along the path.
    d. "Drag"; constant force to slow down the particles and keeping the particles on the path.

Adding VFX to the curve [read]

Now that we have our curve, we are free to add as many VFXs as we want. There's still some technical know-how that needs to be performed to get the curve system working, which we will explain now.

1. Locate the FX you want to add under Realistic VFX Pack/Prefabs/Curve FX.
2. Drag-and-drop the FX as a child of "VFX (add vfx as child of me)" under the "Curve" prefab.

3. **[Important]** To refresh the curve, you need to select the root "Curve" prefab once more. Now the particles of the FX should follow the curve. Now's a good time to play around with the curve properties to get a desired movement.

(*):

The children of "Points [DO NOT TOUCH]" are the points that make up your path. If you need higher precision than the 'Curve' editor allows you, you can choose to move the points directly. **Remember**; you need to update the curve by selecting the root "Curve" gameobject, otherwise your changes will not take effect.

Object Avoidance (**Should Read**)

Here we will show you how to set-up obstacle avoidance for the VFX graphs that allow it. Keep in mind; to achieve object avoidance on the GPU that we have implemented, some limits are necessary. Only **ONE** obstacle can be avoided — for more, you need to combine the meshes yourself and do the following steps with the joint mesh instead.

Creating the obstacle SDF [read]

First of all, we need to bake the SDF that will represent the obstacle on the GPU side.

1. On the top navbar on the Unity Editor; Go to Window/Visual Effects/Utilities/SDF Bake Tool.
2. Assign the Mesh or Mesh Prefab (change the model source to Mesh Prefab to use a mesh prefab).
3. In the preview, you will see the bounding box. Make sure the bounding box covers the entire mesh. You can update the bounding box by pressing the "Fit box to Mesh" button.
4. Click "Save SDF" and select a folder in the Unity Project.

Object Avoidance [read]

Now that we have our SDF, we can make our VFX avoid it. Go to any VFX that has the 'Object Avoidance' category, e.g. the "Butterfly_Flock".

1. Assign our newly baked SDF to the 'Signed Distance Field of Mesh' property under the 'Object Avoidance' category in the VFX.
2. Make sure 'Show Property Gizmos' in the Visual Effect component is **on** and that the 'Edit Gizmo' button is **selected** on the 'Mesh Bounding Box' property.
3. Now we can align the 'Mesh Bounding Box' to the object that we are avoiding. To do this, you copy the **world** position, rotation and scale of the object and assign the values to the 'Mesh Bounding Box' property in the VFX. Some changes may be needed. If the Gizmos of the bounding box is showing, you can see and change the box to align with the *bounding box*(**) of the object you want to avoid.
4. Make sure the Object Avoidance's property 'Enabled' is **on**.

5. The other two properties under the 'Object Avoidance' category is 'Avoidance Distance'; minimum distance the particles will try to be away from the object. 'Avoidance Force'; the force applied to the particles to try to keep that minimum distance.

(**):

A bounding box is a box aligned with some object and fully enclosing that object.

FAQ + Known Problems

Problems & Solutions

Unity is an engine with multiple versions. Different versions may encounter unique problems. Here we will try to answer some of the regular problems that may appear when using this package. For problems that this thread does not answer, please contact us at: saitamastudiobusiness@gmail.com

[problem]   **I can't edit the properties in the VFX**

[answer]   Sometimes when a Visual Effect is under a prefab it does not allow you to edit the VFX directly in the editor. Try to "Unpack" the prefab (right-click the parent prefab of the VFX and press "Prefab/Unpack").

FAQ

[question]   **What is customizable? Can you change colors & textures?**

[answer]   Everything is customizable. So yes, you can change the colors, textures or any other properties such as movement, spawn area, spawn rate etc.

[question]   **If you attach a VFX to an object and move it around, does it follow it?**

[answer]   Yes. All VFX are in world space, so the effects will follow the object.

[question]   **Where did you get the models for the Demo Scenes?**

[answer]   The environment, demo models, and other assets are custom-made. The 'Showcase2' demo scene includes some assets from Unity Technologies' free-to-use collection. You are free to use these and all other demo scenes in any project, whether commercial or non-commercial.

[question]   **Does the pack include Shuriken based particle system (previous Particle System)?**

[answer]   Unfortunately, the pack doesn't include VFX from the Shuriken particle system, as we needed the advanced features of VFX Graph for object avoidance, particle curves, smooth falling leaves etc.

[question] **What's the difference between this Pack and the previous 'Essentials Environment VFX Pack'?**

[answer] This pack focuses more on realism, targeting games & projects with realistic graphics. It includes custom features like object avoidance, advanced leaf fall dynamics, and a key highlight: a curve editor for precise particle movement along any path. Due to the increased complexity and higher quality, the price reflects the added value.