



User Manual

Created by Arthur Grohe, last edit: Do, 24. Dezember 2020

Contents

Quick Start.....	2
1. Find the [GOPainter] Prefab.....	2
2. Drag [GOPainter] into your scene.....	2
3. Setup some GameObjects that you want to paint with.....	3
4. Activate GOPainter and start painting!.....	4
Painting objects on a terrain.....	4
Properties and components of GOPainter.....	5
A. Properties.....	5
B. Components.....	5
- GO Randomizer.....	5
- GO Surface Snapper.....	6
- GO Switcher.....	6
- Object Paint Animation.....	7
- Object Factory.....	7
- Other Components.....	7
Hotkeys / Shortcuts list.....	7
Simulate Physics (Gravity).....	9
Customizing.....	10
List of GOP events: (GOPainter.cs, Line 197 - 223).....	10
Known issues.....	11

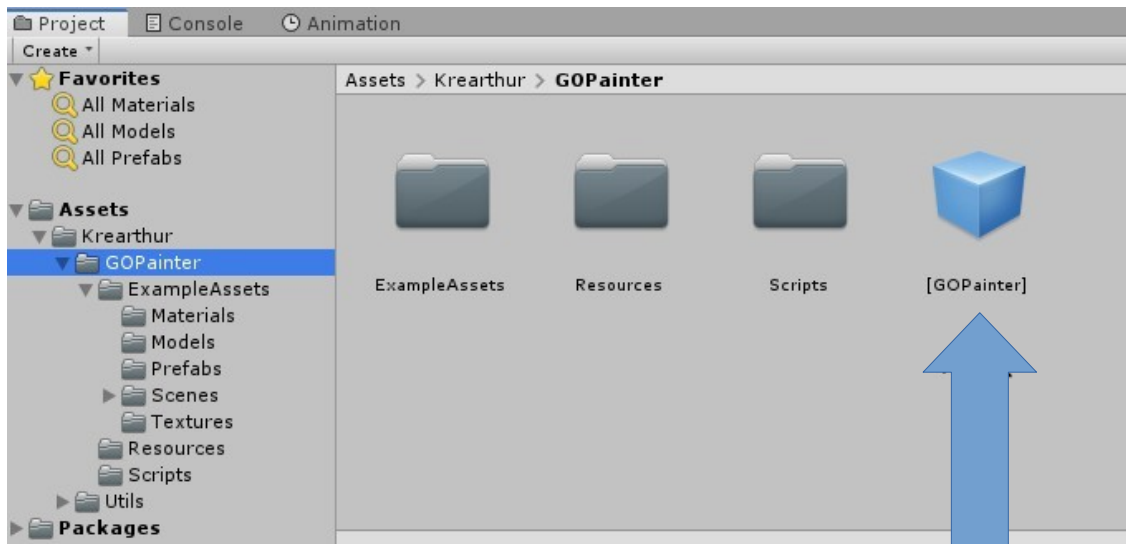
If you have questions, found bugs or have other feedback you can create an issue at <https://github.com/krearthur/gopainter-feedback/issues>, but I recommend watching the [tutorial video](#) and reading this short manual first ;)

The tool was tested with Unity 2019.2.x, 2019.4.x and 2020.1.x

Quick Start

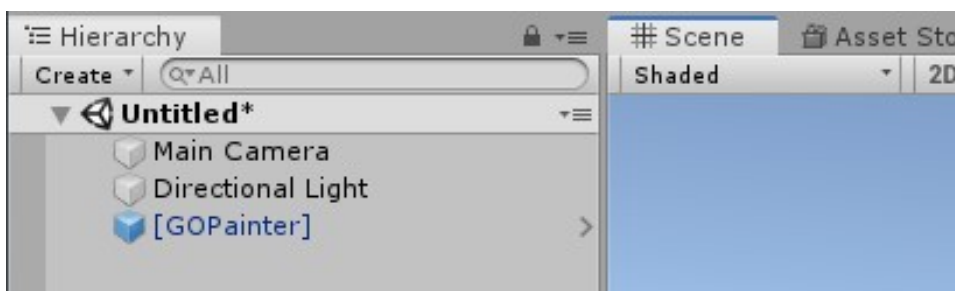
1. Find the [GOPainter] Prefab

After importing the package from Asset Store, look into Assets/Krearthur/GOPainter. There you should find the [GOPainter] Prefab:



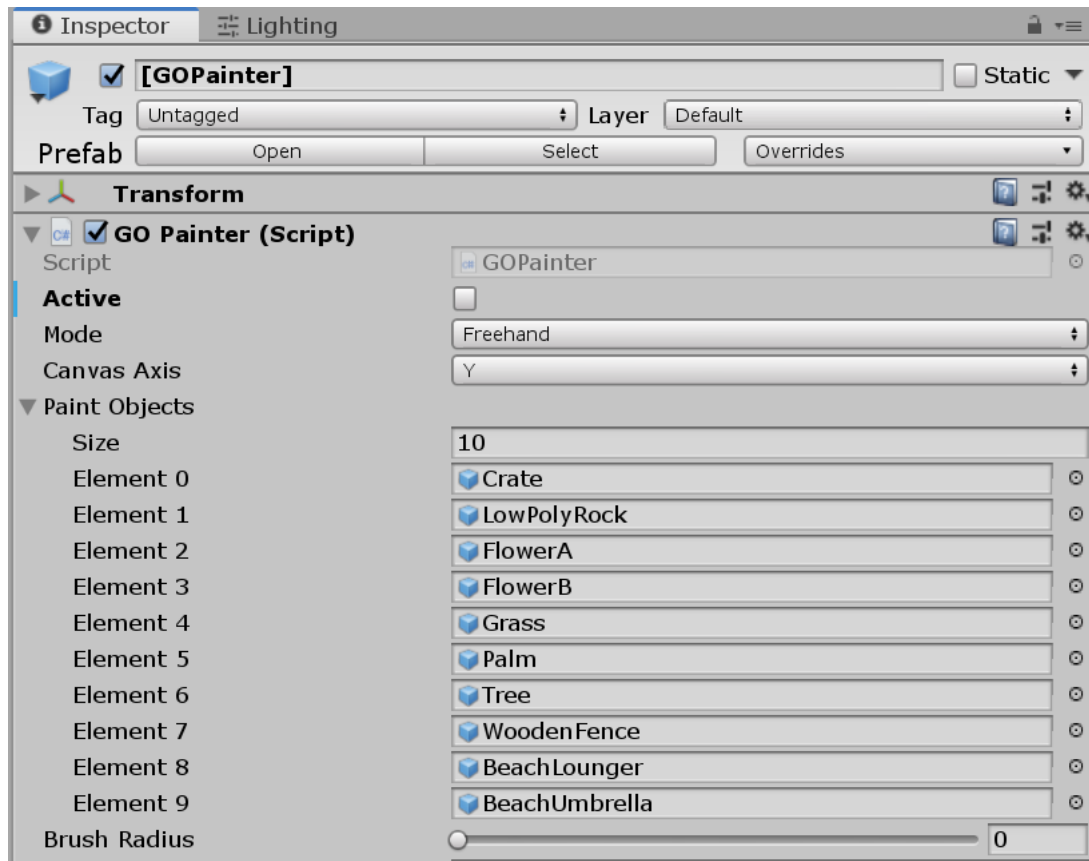
2. Drag [GOPainter] into your scene

After you dragged the prefab into your scene it could look like this:

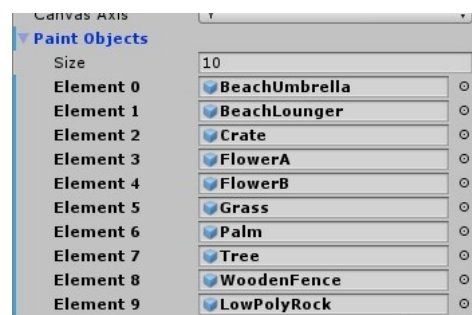


3. Setup some GameObjects that you want to paint with

Click on the GOPainter object in the **Hierarchy**, so that the **Inspector** shows its components:



Now you can set up to 10 GameObjects, either Prefabs, or any other GameObject, into the *Paint Objects* field in GOPainter. Just drag and drop your GameObjects into that list.

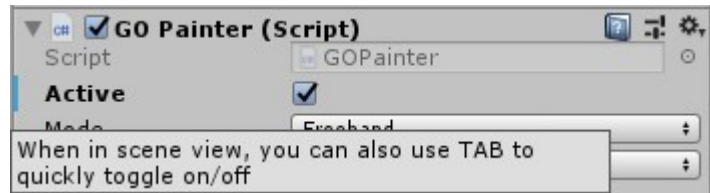


If you want to remove an object from the list, click on it then press the Delete key.

4. Activate GOPainter and start painting!

Now you're all set and can go painting.

To activate it, click the *Active* Checkbox, or, if you click into the **Scene** view, you can press the tabulator key to quickly toggle activation of GOPainter.



Painting objects on a terrain

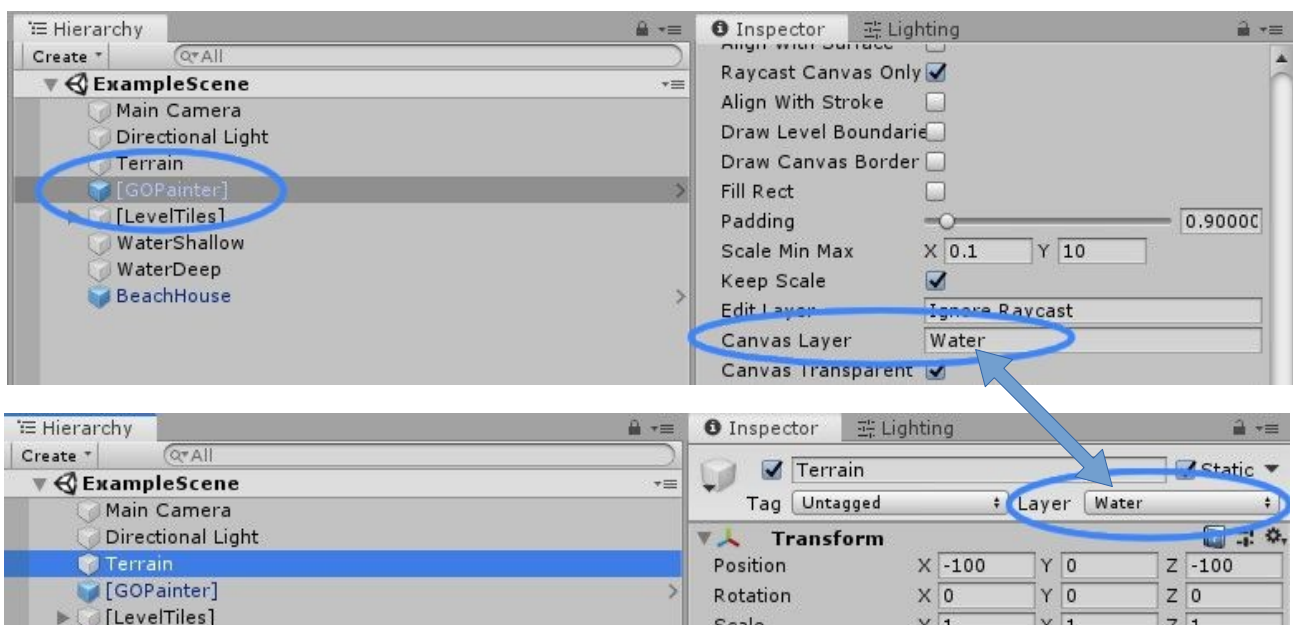
Since version 1.3 you can paint on terrains without manual config, because a new option was added:

Auto Set Terrain Layer ☒

When this is checked, the scene is scanned for terrains and then their Layer is set to the value of GOPainters *Canvas Layer* property.

For versions less than 1.3, here is the setup:

To make a terrain your painting canvas, set the terrains' *Layer* the same as the configured *Canvas Layer* in GOPainter:

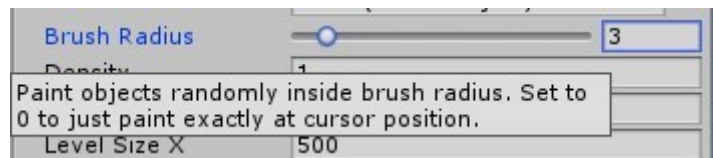


That's all you need to paint on terrains.

Properties and components of GOPainter

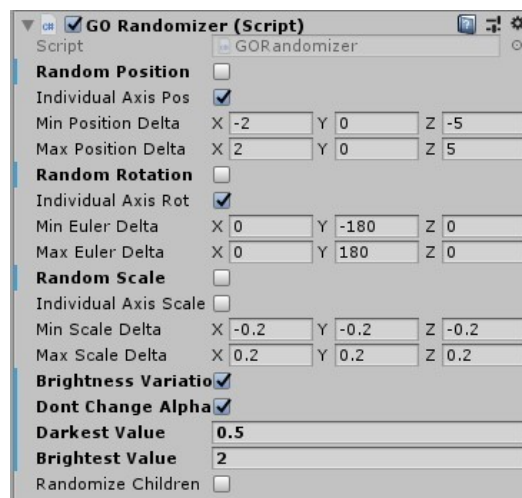
A. Properties

You can hover over every property of the GOPainter object with the mouse and a useful tooltip will appear. E.g.



B. Components

◆ GO Randomizer



The GO Randomizer can randomize position, rotation, scale or brightness of painted objects. Just enable this component to be active. You can enable each checkbox as needed.

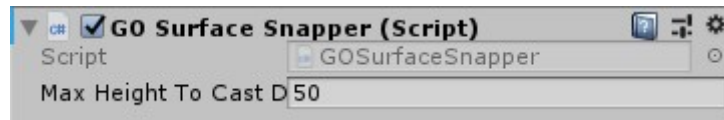
The ...*Delta* properties mean that an objects transform property will be added or subtracted by a random value between the *Min* and *Max Delta* **relative** to the current transform property values.

Careful with the **Brightness Variation**: It works that way: Every time the shaders are reloaded, the brightness of the materials color will randomly change. That is when the level is loaded, or when you press play in the editor or leave play from editor.

Note: If you change *Darkest Value* or *Brightest Value* it will be applied to every object that was painted while the *Brightness Variation* checkbox was checked and GO Randomizer was enabled.

For example: You paint some rocks with brightness variation between 0.8 – 1.2. After that you change the variation to 0.5 – 1.5 for some other objects. Now when you hit play also the rocks will change to that new variation value. ... Sorry!

◆ GO Surface Snapper



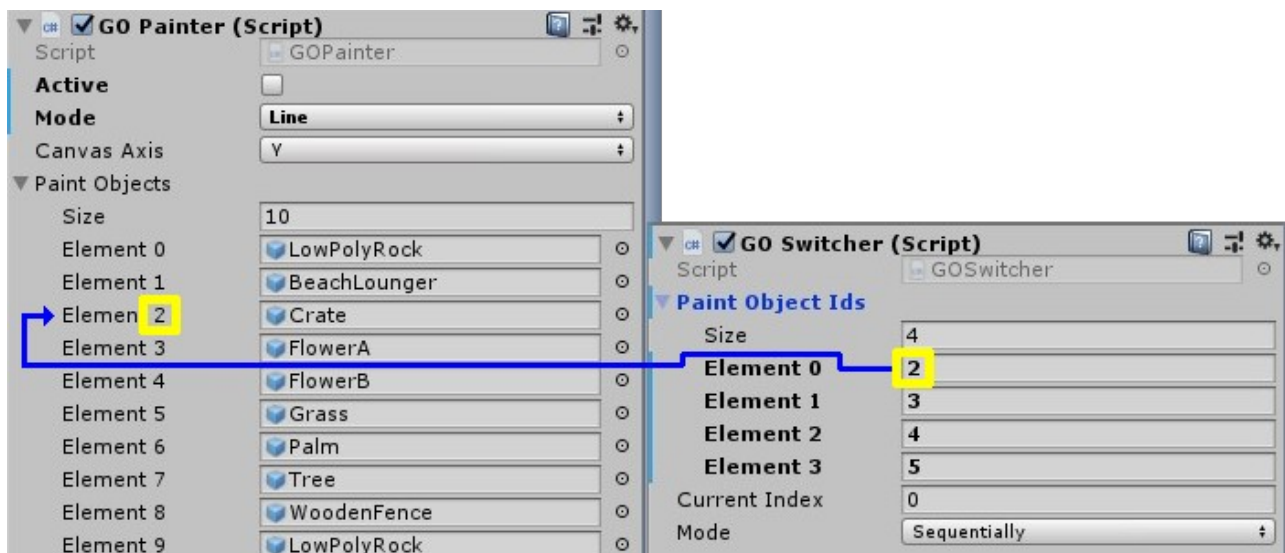
The Surface Snapper extends the surface snapping feature to work on uneven terrains for all modes, e.g. so you can draw a line of rocks over a hill, and all rocks will snap to the surface of the hill.

When you disable the surface snapper, the rocks would go through the hill in a straight line.

Of course this is sometimes wanted, e.g. to paint a bridge over a cliff. You can always disable this component if you don't need it.

Max Height To Cast Down: Before snapping down to the ground, the object is virtually put up x meters and then cast down.

◆ GO Switcher



GO Switcher automatically switches between a pool of objects, set in the *Paint Object Ids* list. Put in the ids (*Element n*) from the *Paint Objects* list of GOPainter of the objects you want to automatically switch between.

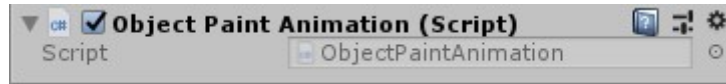
Current Index: Shows where the current index is. You can change that manually if you want to reset the index for example.

Mode: Either switch sequentially from first to last object or switch randomly.

Tip: You can also set the same id multiple times, for example: You have FlowerA, FlowerB, and Grass. You set the mode to Randomly. Now if you want to have more grass than flowers, you could set the list to a size of 10 elements. Then the first 5 elements you put in the Id for the grass object, and for the last 5 elements you put in two times FlowerA, and three times FlowerB. Now if you

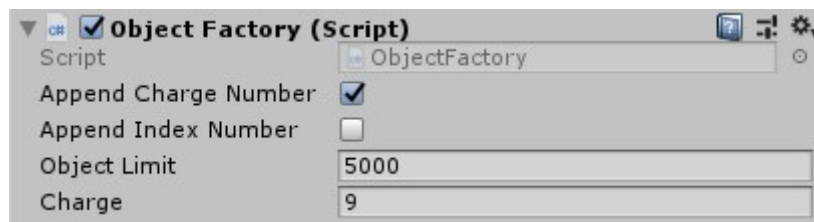
paint, you have a probability of 50% to paint grass, 20% to paint FlowerA and 30% to paint FlowerB.

◆ Object Paint Animation



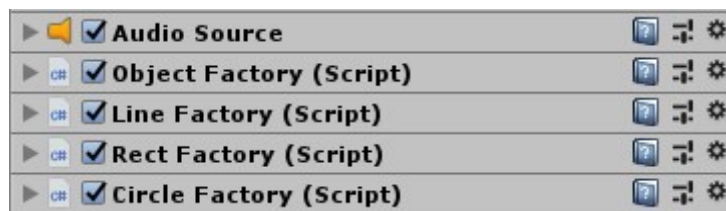
This component is responsible for the nice popping animation when placing an object. If you don't want the animation, just disable this component.

◆ Object Factory



This is mostly a base service class for GOP, but you can modify minor things here. Just hover over the fields to get an explanation.



◆ Other Components





The other components work as service classes for GOP and should not be modified. You can always collapse them to save space.

Hotkeys / Shortcuts list

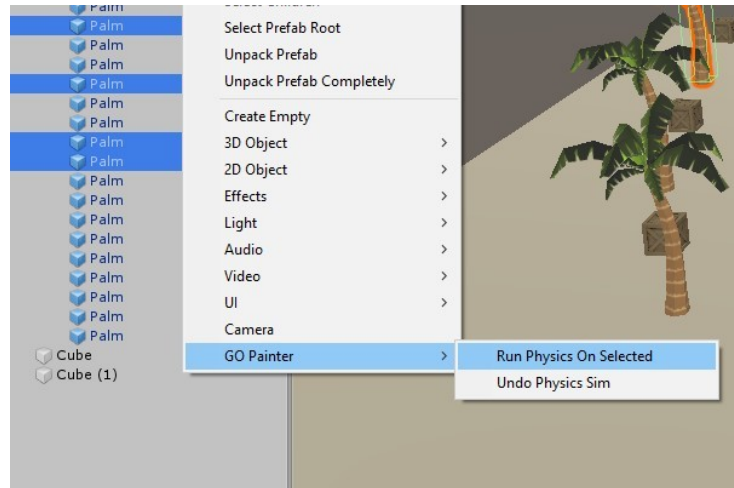
GO Painter is designed for a quick workflow so it has many shortcuts.

Shortcut	Effect
Tab	Activate / Deactivate GOPainter
Esc	Deactivate GOPainter, cancel scale mode
Space	Change paint canvas axis, it cycles from X -> Y -> Z -> X ...
1 ... 0 (horizontal number keys on keyboard)	Switch GameObject
Q	 Freehand mode
W	 Line mode

E	 Rectangle mode. Press again to switch between filled rectangle or border mode
R	 Circle mode
Ctrl+Q or +W or +E or +R	Delete with the current mode. In Freehand mode you can either click on any object, or click and drag to delete multiple objects. The deletion with circle tool works the same as with rect tool.
A	Cycle between Align to Stroke -> Align to Surface -> Don't align -> Align to Stroke ...
S	Scale mode. Move mouse horizontally to scale. Click or S to submit, Esc or Right-Click to cancel.
D	Toggle dynamic grid.
F	Focus to object (standard Unity feature, but very useful for this tool)
G	Toggle Snapping to Grid
Z	Toggle Visibility of Paint Canvas. When visible, only objects above it can be deleted with the deletion methods.
Left click	Paint object
Ctrl + left click	Delete object on cursor
Right click	Pick object. Now you can paint with it.
Alt + right click	Pick object and move paint canvas to that position.
Ctrl + right click	Grab-pick object
Mouse Scroll	Default Unity zoom function. But when in either Line, Rect or Circle mode and while dragging, scrolling changes the <i>Padding</i> between objects.
Alt + Mouse Scroll	Rotate Object around its Up-Axis. Rotation amount is configurable via the <i>Rotation Step</i> Property.
Ctrl + Mouse Scroll	Change <i>Brush Radius</i> . Only has effect in Freehand mode.
Shift + Mouse Scroll	When not in Circle mode: Change position of paint canvas along the current paint canvas axis. In Circle mode: Change the arc degrees.

Simulate Physics (Gravity)

You can select multiple objects with ctrl-click and then right-click in the **Hierarchy** or go to GameObject/GOPainter, to open the context menu. There you will find a GOPainter menu item:



Click on *Run Physics On Selected* to simulate gravity on all selected objects. If some of them don't have a collider or rigidbody, they will be added to them. The simulation runs maximum of 1000 steps, which is enough for most cases. After the simulation the added components (if any) will be removed, so the objects are the same as before.

To undo that, open the menu, no matter if you selected the objects, and click *Undo Physics Sim*.

Credit for the original physics sim code goes to Sebastian Lague.

[<https://www.youtube.com/watch?v=SnhfcdtGM2E>]

Customizing

If you want to customize GOPainter further to your needs, you can just edit the source code, since I included it completely.

But be aware, the code is not the cleanest of all.

Another „easy“ way to add new features is using the *Event System* I created. Just look at the other components like GOSwitcher and GORandomizer, that make use of it. For example you can register a callback function for an event that fires after an *GameObject* is painted:

```
public void Register()
{
    if (goPainter == null)
    {
        goPainter = GetComponent<GOPainter>();
    }
    goPainter.OnObjectPainted += RandomizeTransform;
}
```

Listing 1 – Register a listener

Here we register the method *RandomizeTransform* as a callback function for the *OnObjectPainted* event.

Next we have to implement this method and can do whatever we want:

```
void RandomizeTransform(GameObject go, Vector3 axis, bool snapToGrid)
{
    if (!enabled) return;
    // Do something, like manipulate the game object...
}
```

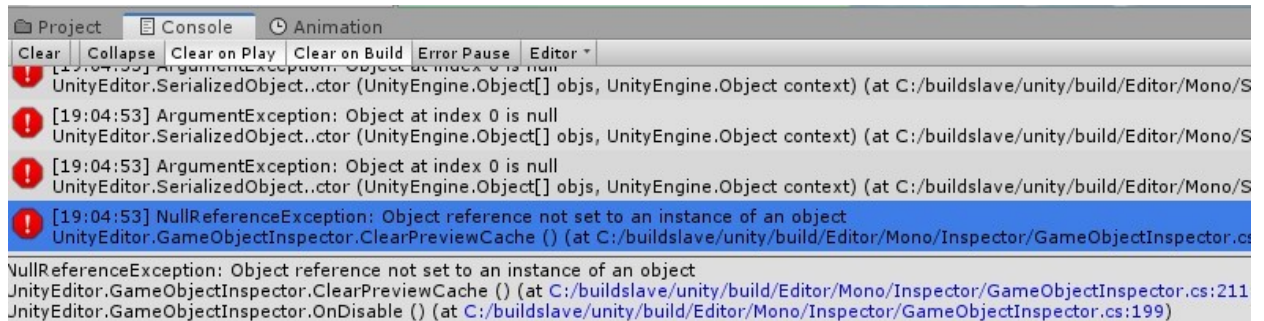
Listing 2 – Implement listener method

List of GOP events: (GOPainter.cs, Line 197 - 223)

```
public event Action OnGOPainterActivated;
public event Action OnGOPainterDeactivated;
/// Called when the holded object's position is changed.
public event Action<GameObject, Vector3, bool> OnHoldingObjectPositionChanged;
/// Called when the holded object is painted into scene.
/// GameObject: The object that is painted.
/// Vector3: The current painting axis.
/// Bool: Snap to grid
public event Action<GameObject, Vector3, bool> OnObjectPainted;
/// Called when objects were mass painted into scene, via Line, Rect or Circle Tool
public event Action<GameObject[], Vector3, bool> OnObjectMassPainted;
public event Action<GameObject[], Vector3, bool> OnObjectMassPaintedLate;
public event Action<GameObject> OnObjectPicked;
/// Called when objects were mass deleted from scene (in fact they just get
/// deactivated), via Line, Rect or Circle Tool
public event Action<GameObject[], Vector3, bool> OnObjectMassDeleted;
/// Called when an object is deleted from scene (in fact it gets deactivated).
public event Action<GameObject> OnObjectDeleted;
```

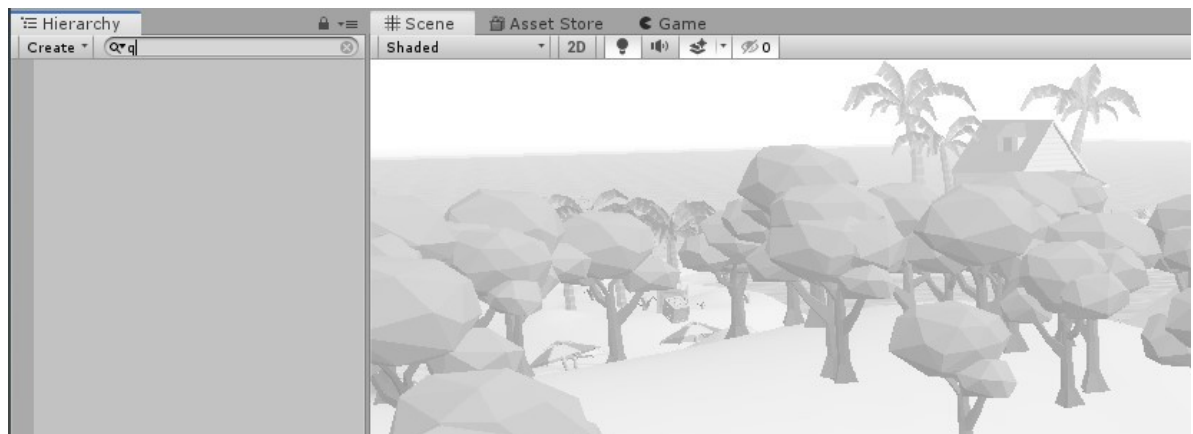
Known issues

- In Unity 2019: When GOP is active and you're compiling a script, there may occur some errors in the console:



Don't worry, this is a [known issue](#) from Unity and has no effect

- In some cases when you're in the scene view and press a letter key, the filter function of the Hierarchy will be used instead of the GOPainter function. This is because the Hierarchy view has the current 'focus' instead the Scene view:



Just 'x' out the filter in the Hierarchy, click in the Scene view and GOP features should work as usual.

Thank you for reading and I hope you have fun using this tool! :)

Kind regards

Arthur Grohe

