Impostors - Runtime Optimization

Documentation

Online documentation:
https://docs.google.com/document/d/1hoP7m5lvUSQYHzlAz6Hwr4DeooZOhYBylGPXxyEgcDY/edit?usp=sharing

Forum:
https://forum.unity.com/threads/wip-impostors-2019-optimization.759110/

Table of contents

Dependencies

IMPORTANT: package works only with Unity 2019.3+. It uses the new mesh API introduced in this unity release.

Minimum required versions. Latest versions are also supported and tested.

Package Dependencies:

- Jobs 0.1.1 - preview
    - Collections 0.1.1 - preview
    - Mathematics 1.1.0
- Burst 1.1.2

Known issues

1. For now it is not tested with HDRP.
2. There might be issues with HDR. If so, please contact me with details.
3. Deferred rendering supported, but Impostors will be rendered in Forward path anyway. If it causes problems, then contact me with details.
4. Built-in Render Pipeline with Scheduled impostors rendering in VR projects doesn't work properly and may cause issues. Fix: use Immediately impostors rendering.
5. Universal Render Pipeline in VR projects may not work properly.

Installation

After downloading and importing asset from the Asset Store you are good to go.
If you have troubles or compilation errors, make sure that the importing process included upm dependencies:

- Jobs
- Burst
- Mathematics
- Collections

Also you can:

Delete “Example” folder

You are safe to delete the “Impostors/Example” folder if you don’t need it anymore.

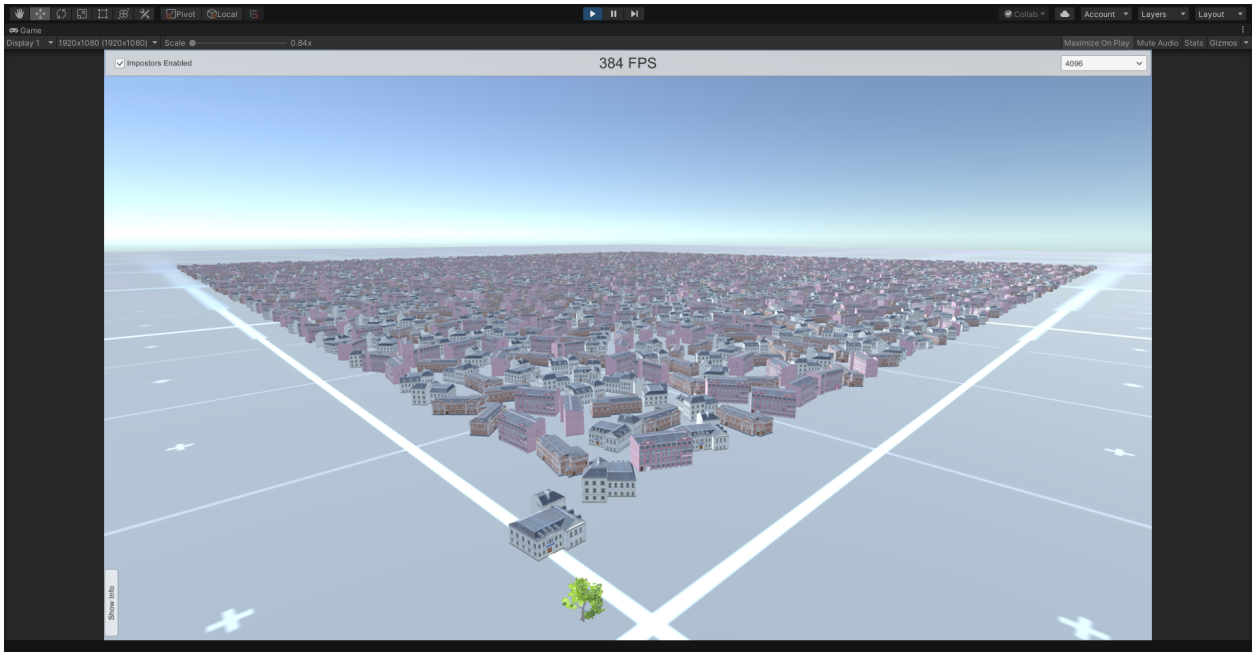
Import as embedded package

If you want you can import Impostors as the upm-embedded package:
https://docs.unity3d.com/Manual/upm-embed.html

How to use

Example scene

Open scene at “../Impostors/Example/Scenes/example.scene” and go into play mode.
You don’t need to do any additional setup for the example scene.
Use top-right dropdown to spawn objects.
Use top-left toggle to enable/disable Impostors.

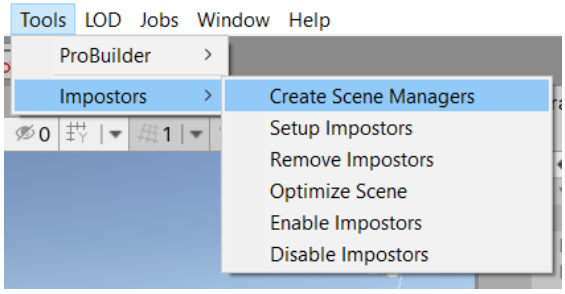


Verify everything works.
If you have a problem on this step, please contact me and we will solve the issue if any.

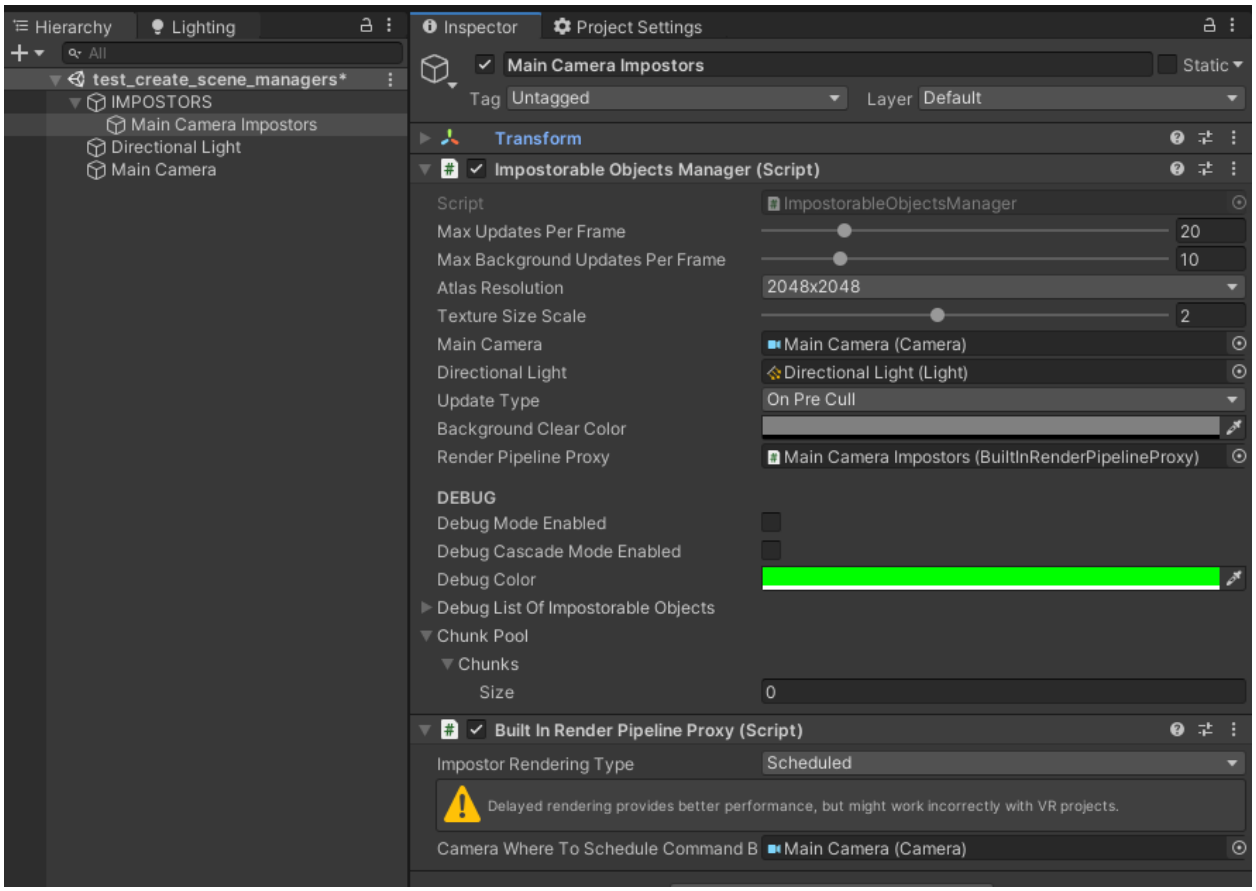
Setup your scene

IMPORTANT! Please, make a back-up before proceeding with this step.

Use “Tools/Impostors/Create Scene Managers” to set up your scene to use impostors.



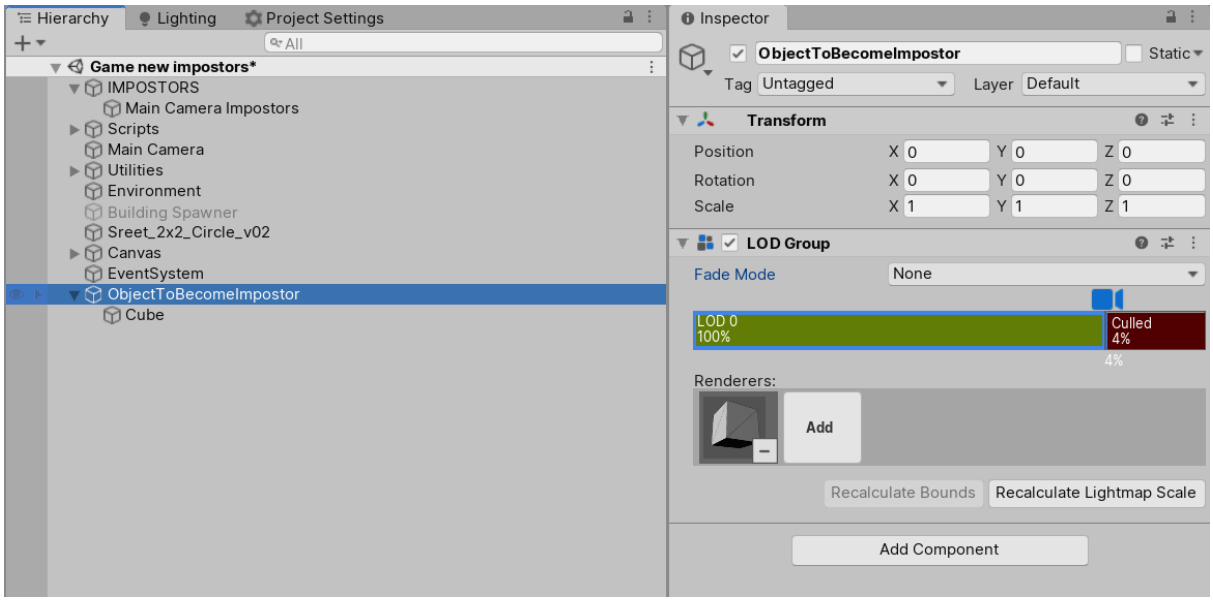
This will create a new game object “IMPOSTORS” at the top of your scene hierarchy. Under this game object will be one child, which setups the main camera to render impostors. If you want more cameras to render impostors, then just duplicate this object and set another camera.



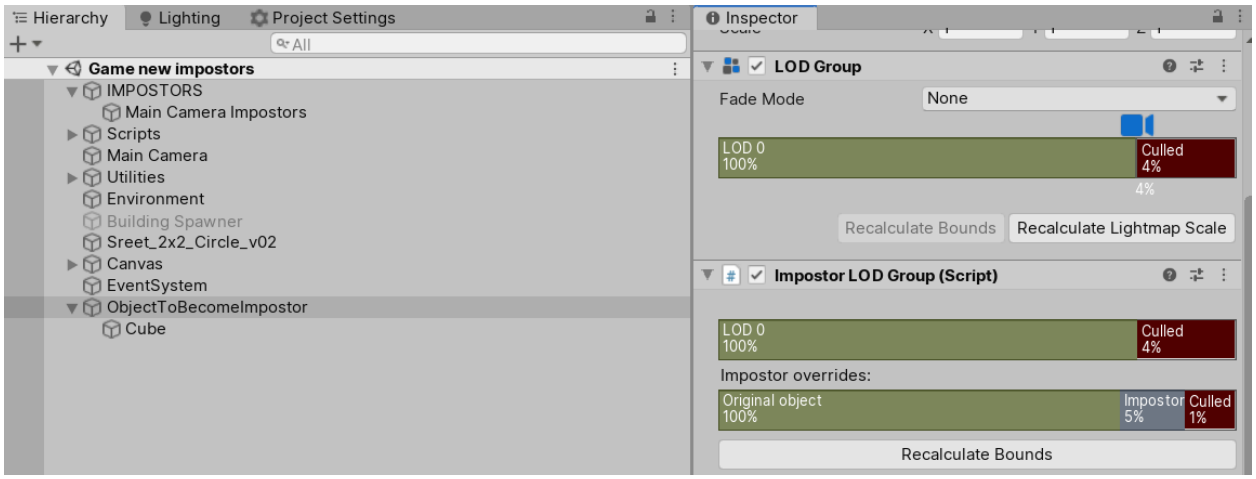
Make sure the RenderPipelineProxy is referenced in ImpostorableObjectsManager.

GameObject setup

After scene setup, you need to mark objects you want to render as impostors with the **ImpostorLODGroup** component.
This component **requires a built-in LODGroup component** on the game object to work.
Your LODGroup may contain only one LOD. For example, this setup is fine:



If your object has a LODGroup component, then use “Tools/Impostors/Setup Impostors” to automatically handle setup of the ImpostorLODGroup component.
After this command gameObject’s inspector must look like this:

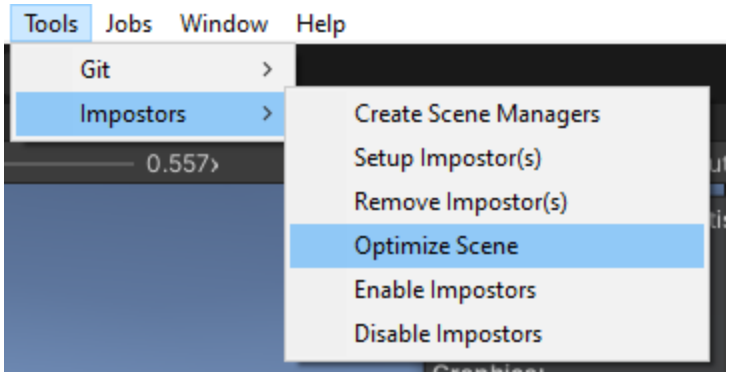


You can change properties as you want. But these properties are cached in OnEnable(), so changes are not reflected at runtime.

Now run the scene and the object will become impostor at some distance.

Other menu items

Under Tools/Impostors/ you can find helpful operations and shortcuts.



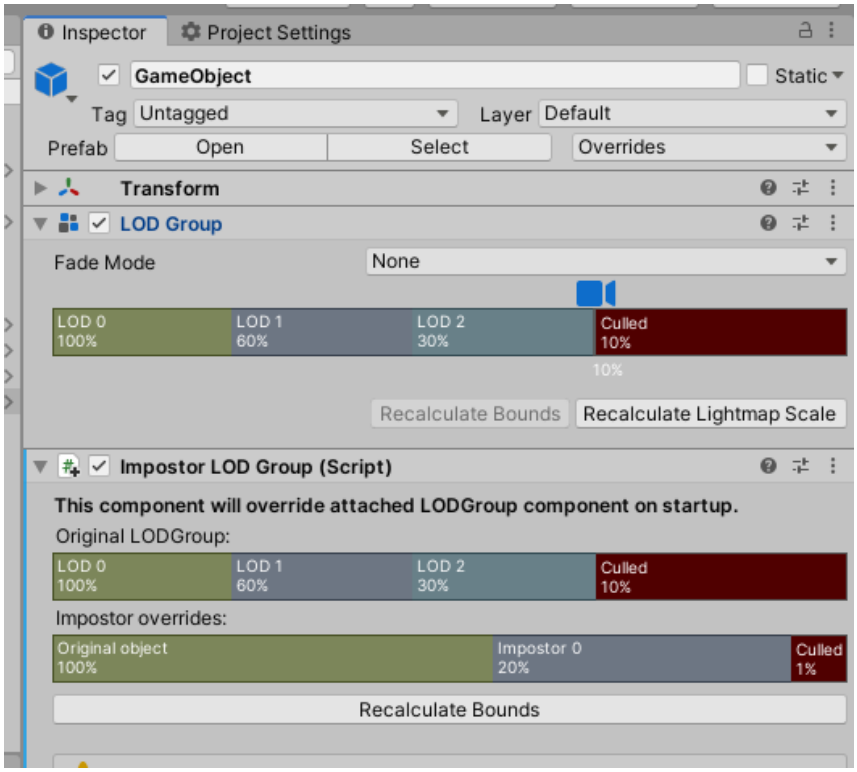
1. Create Scene Managers - creates managers that are required to Impostors to function.
2. Setup Impostor(s) - you can select one or multiple game objects in hierarchy, and this command will try to automatically set up the ImpostorLODGroup components onto selected objects. Note: objects must have unity's built-in LODGroup component to work properly with this command.
3. Remove Impostor(s) - opposite command that removes all Impostors components from selected objects.
4. Enable Impostors - enables all ImpostorLODGroup components in the game.
5. Disable Impostors - opposite command that disables all ImpostorLODGroup components in the game. Useful when you want to look at your scene without Impostors.

Optimize Scene menu item

I moved this command in it's own section because it requires a lot of explanations.
In a nutshell, it finds all LODGroup components in the current scene and applies the "Setup Impostor(s)" command on each object.
Warning: this command is quite heavy and might take a lot of time. For example on 10k objects it may take 20 seconds. But don't be afraid, this setup process is required only once and doesn't impact runtime performance.

Use at play mode
This command is **designed to be used at runtime**, when you are running your scene. This way is preferable, because you can undo all setup operations just by stopping the play mode and prevent any chances to corrupt your scene.
This is handy when you want to quickly look at what Impostors can do with your scene.
Warning: this command cannot be used in play mode with objects that use static batching. Objects marked as static will combine their meshes at scene startup. There is no way to access original mesh data to render impostors.

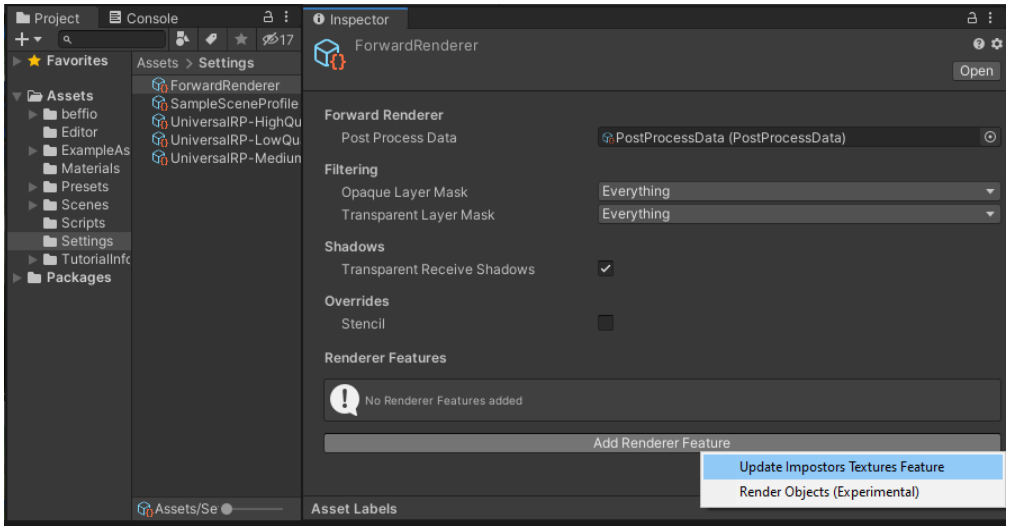
Be careful with prefabs
In most cases you have your scene built-up with lots of prefabs, and running "Optimize Scene" command at edit mode will add an override component on each prefab instance, like this:



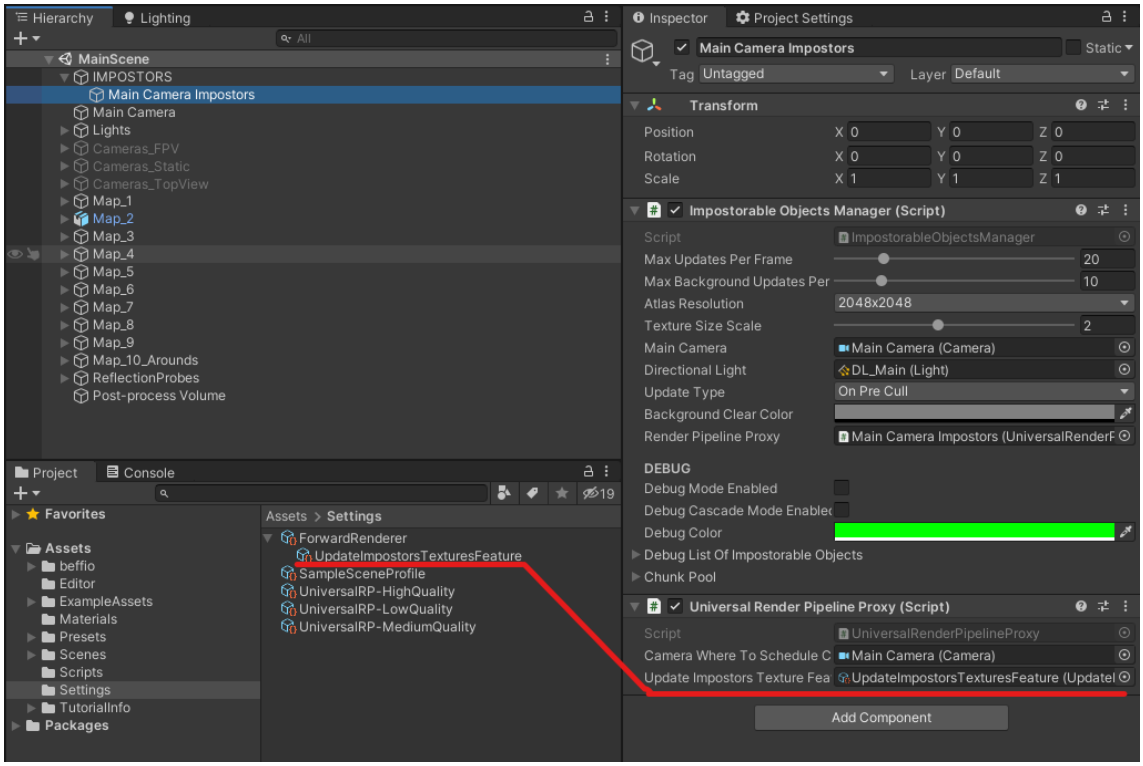
Which is quite bad if you then decide to apply the ImpostorLODGroup component to prefab.

Universal Render Pipeline

Impostors support URP but require some additional setup.
Select your Renderer Data asset. By default it is at path “Assets/Settings/ForwardRenderer”.
Click on “Add Renderer Feature” and select “Update Impostors Textures Feature”.



Then in scene select “ImpostorableObjectsManager”. Make sure there is a “UniversalRenderPipelineProxy” component, and assign feature to corresponding field:



Best practices

Mobile devices

Mobile GPUs are very different from standalone GPUs, so it requires additional know-hows on setting up your project using the Impostors package.

1. Don’t use HDR impostors if you can.
2. Don’t use Atlas Resolution bigger than 2048. 1024x1024 provides best performance on mobile devices.
3. Limit MaxUpdatesPerFrame to something like 20, not more.
4. It’s better to use TextureSizeScale = 1. Or you can use values below 1 to save some memory. You can use debugCascadesModeEnabled on device to investigate impostors’ texture sizes.
5. On mobile devices scheduled impostors rendering is crucial to achieving max performance.
6. In my tests and on my devices I got better performance using OpenGLES3 GraphicsApi in comparison to Vulkan. It depends on your project and devices you are targeting so it worsts testing.

Manual update mode

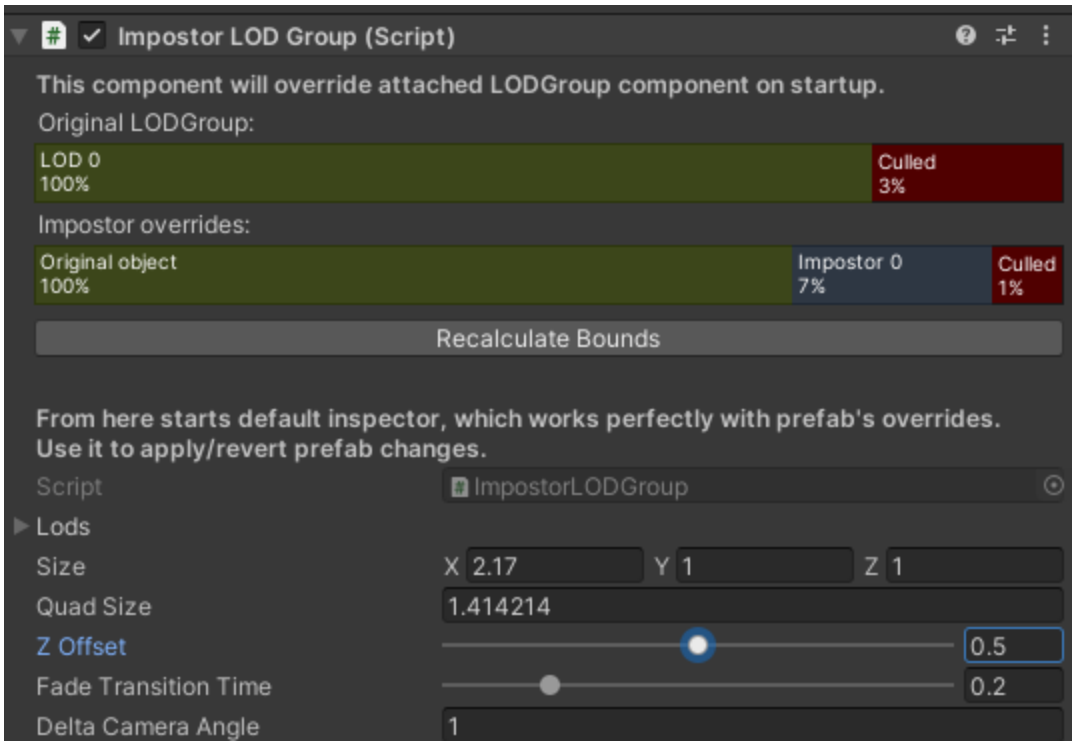
You can explicitly control when to run the impostor system and where to draw impostors.
Here is the example script:

```
using Impostors.Managers;
using UnityEngine;


public class ManualImpostorsUpdate : MonoBehaviour
{
    [SerializeField]
    public int _skipFrames = 1;

    [SerializeField]
    private ImpostorableObjectsManager _impostorableObjectsManager = default;

    void LateUpdate()
    {
        if (Time.frameCount % (_skipFrames + 1) == 0)
        {
            _impostorableObjectsManager.UpdateImpostorSystem();
        }

        _impostorableObjectsManager.DrawImpostorsForCamera(Camera.main);
    }
}
```

FAQ

Bottom part of the impostor is under the ground. How to fix?

It’s a common issue and fix is very simple. Exit play-mode, go to your object with ImpostorLODGroup component. Increase Z Offset and try to run the scene again. Problem should be fixed.



How to spawn/destroy objects with ImpostorLODGroup at runtime?

Simply Instantiate()/Destroy() them.

How to setup object to use the ImpostorLODGroup component at runtime?

Take a look at ImpostorsEditorTools.SetupImpostorLODGroupToObject() method.
This is an example on how to properly add and setup ImpostorLODGroup to gameObject at runtime.

Impostors have outlines. How to fix?

By default these outlines are gray and almost invisible at well-lit scenes.
But in dark scenes these gray outlines might be hardly noticeable. You can fix that by changing ImpostorableObjectsManager.BackgroundClearColor to black.

Important: make sure that the alpha channel of color is 0.

Note: after changing this value you will not get instant feedback, you need to move camera and cause impostors updates.