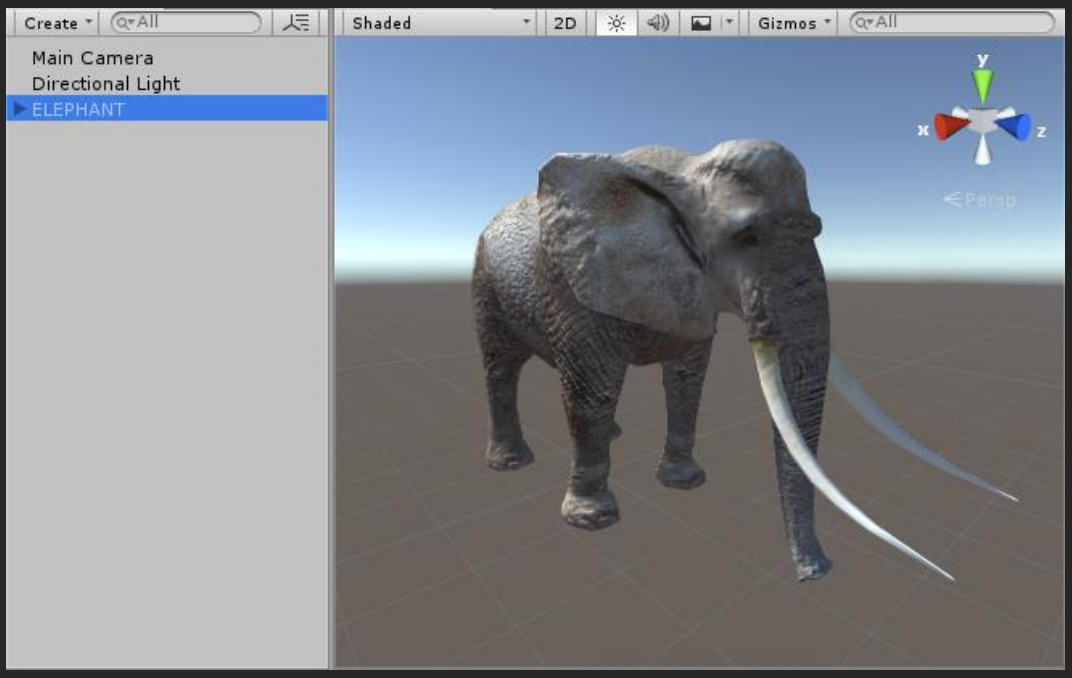
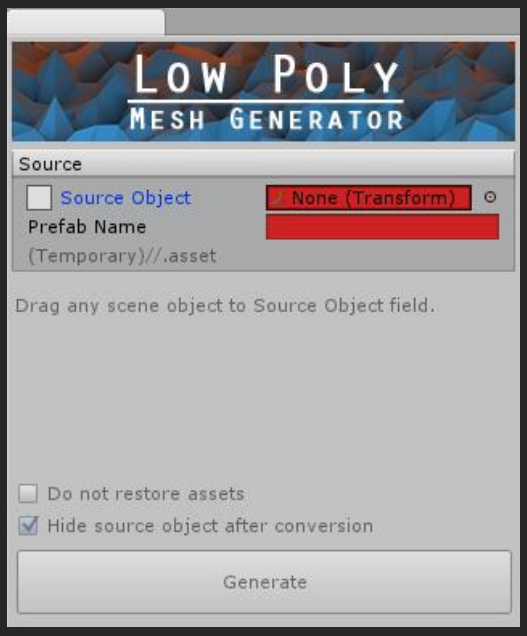
Quick Start

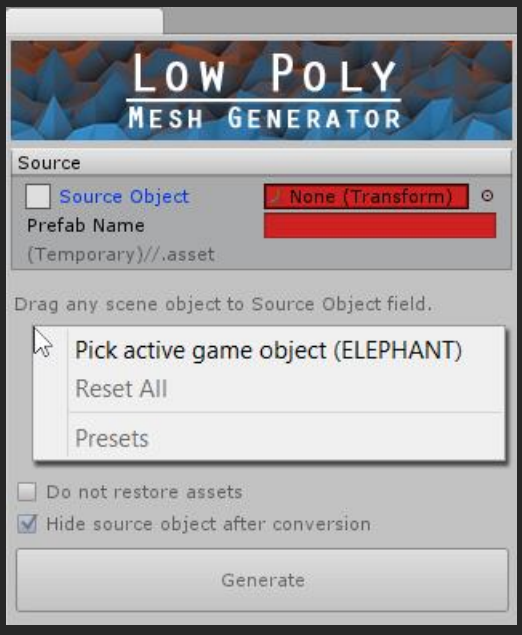
1. Add mesh to the scene.

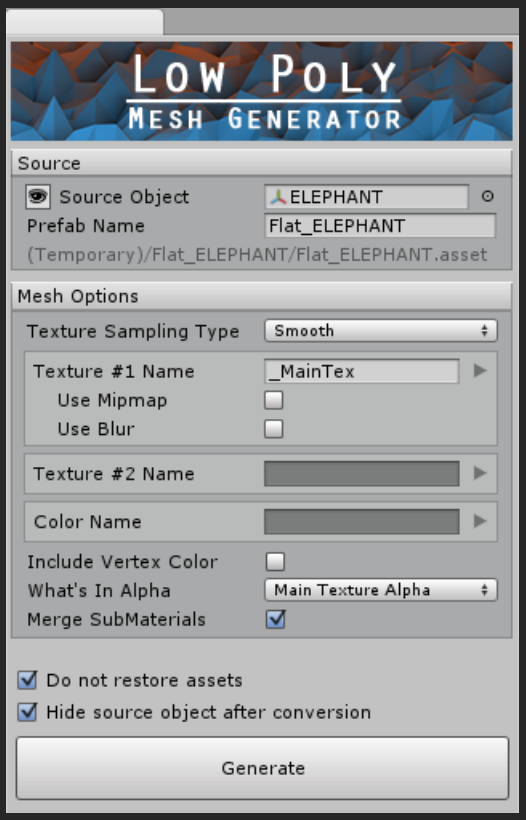




2. Open Low Poly Mesh Generator tool from *Menu -> Window -> VacuumShaders -> Low Poly Mesh Generator*.

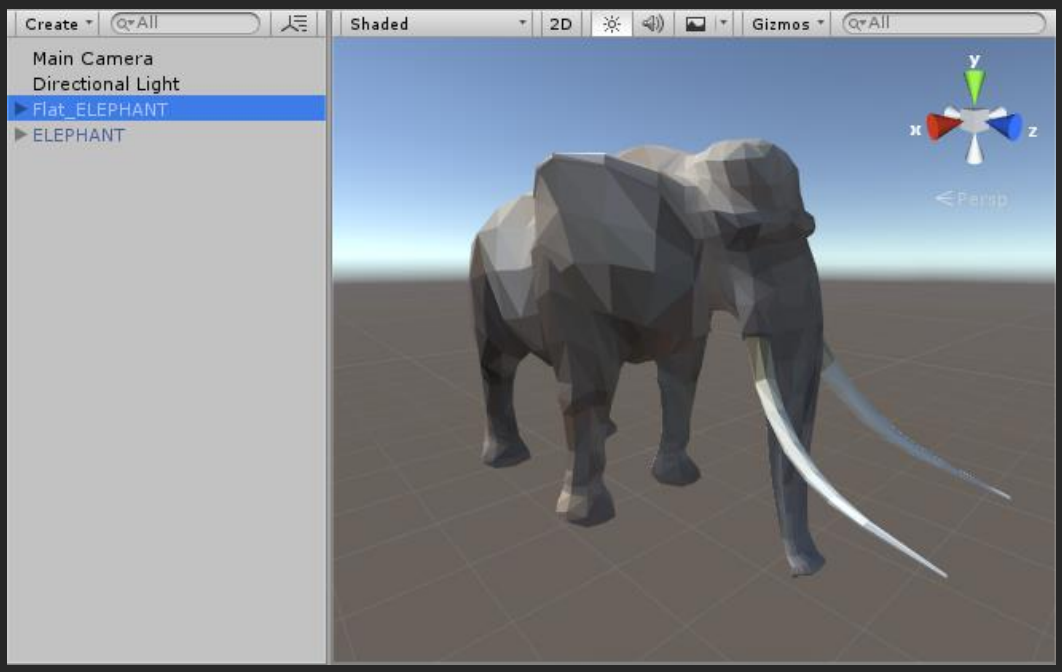
3. Drag mesh into *Source Object* field or pick it using editor window context menu.





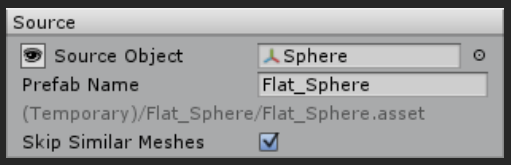
4. Click generate button.

Low Poly Mesh Generator will generate new mesh, bake material texture with name *_MainTex* into mesh vertex color, create prefab with *Prefab Name* with new converted mesh using vertex color shader material and instatiate prefab in the scene at the same position as *Source Object*.



That's all.

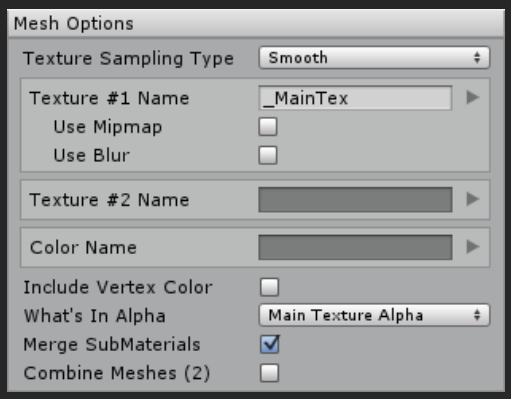
Source



Source Object – Object that will be converted (including all its children in hierarchy).
Prefab Name – This is the name of prefab that will be created inside Assets/(Temporary)/ PREFAB NAME / folder.
Skip Similar Meshes – If *Source Object* contains multiple same meshes (mesh name defines *similarity*) there is no need to convert and save them all. It's enough to convert only one of them and in final prefab it will be used. But in some cases it is necessary to convert all meshes in the *Source Object* – in this case this option must be turned off.

- Button changes *Source Object's* visibility. If *Source Object* is child, than visibility is controlled by its parent object and this button is disabled.

Mesh Options



Texture Sampling Type

- Hard
- Smooth

Texture #1 Name - Texture parameter inside material from where texture data will be read.
Pick-up button displays texture parameters that are available inside material.
To bake texture it must be readable. Check "Read/Write Enabled" inside texture import settings.
Unity readable texture formats are - ARGB32, RGBA32, BGRA32, RGB24, Alpha8 and DXT.

Use Mipmap – Reads texture data from its mipmap, cheap and fast way for achieving blur effect.

Use Blur - GPU accelerated texture blur effect using Gaussian filters. Requires RenderTextures support.

Texture #2 Name (available only if Texture #1 is active) – Bake two textures blending them by **Blend Type** parameter.

Color Name - Color parameter inside material from where color data will be read.
Pick-up button displays color parameters that are available inside material.

Include Vertex Color – If source mesh has vertex color it can be saved within generated mesh.

*Final baked vertex color is calculated by: Texture * Color * Vertex Color.*

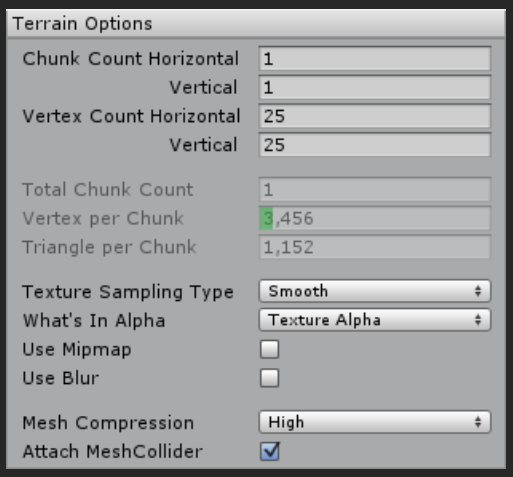
What's In Alpha - Defines what is saved inside vertex color's alpha channel.

Merge SubMaterials – If original mesh uses multiple materials they will be merged into one. Useful for achieving one draw call optimization.

Combine Meshes – If *Source Object* contains multiple meshes they can be combined. Options is available only if final combined mesh vertex count will be less than 65.000.

Mesh Compression - Compressing meshes saves space in the built game, but more compression introduces more artifacts in vertex data (including color). For multi-chunk meshes editor automatically adds *CompressedMeshLoader* script for fixing visible edge artifacts (high compressed meshes) in run-time.

Terrain Options



Chunk Count Horizontal/Vertical – Instead of converting source terrain into one mesh, it can be divided into multiple chunks.
Vertex Count Horizontal/Vertical – Chunk vertex count. Each chunk may have maximum 65,000 vertices.

Texture Sampling Type

- Hard
- Smooth

What's In Alpha - Defines what is saved inside vertex color's alpha channel.

Use Mipmap – Reads texture data from its mipmap, cheap and fast way for achieving blur effect.

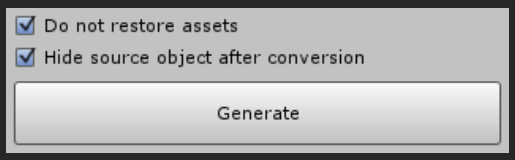
Use Blur - GPU accelerated texture blur effect using Gaussian filters. Requires RenderTextures support.

Mesh Compression - Compressing meshes saves space in the built game, but more compression introduces more artifacts in vertex data (including color). For multi-chunk meshes editor automatically adds *CompressedMeshLoader* script for fixing visible edge artifacts (high compressed meshes) in run-time.

Attach MeshCollider – Generated gameobject will have attached MeshCollider component using the same mesh as MeshFilter. Depending on mesh resolution may be time consuming.

Note:
Terrain texture baking relays on textures used by Unity Terrain system itself.
If terrain system uses custom shader material its textures may be ignored.

Misc



Do not restore assets – If baked textures and meshes are not readable or have unsupported file formats, before conversion Low Poly Mesh Generator changes their settings and forces Unity to reimport them. After conversion file settings are restored and are reimported again. All these steps require much more time then conversion itself. Checking this option speeds up total conversion time minimum twice.

Hide source object after conversion – After conversion new prefab is instantiated in the scene in the same position as the *Source Object*. It's helpful to hide *Source Object* in this case.

Source Object's visibility can be controlled by icon.

Working with presets

Low Poly Mesh Generator editor window settings can be saved as presets for later use.

Presets are accessible from contex (right click) menu of editor window.