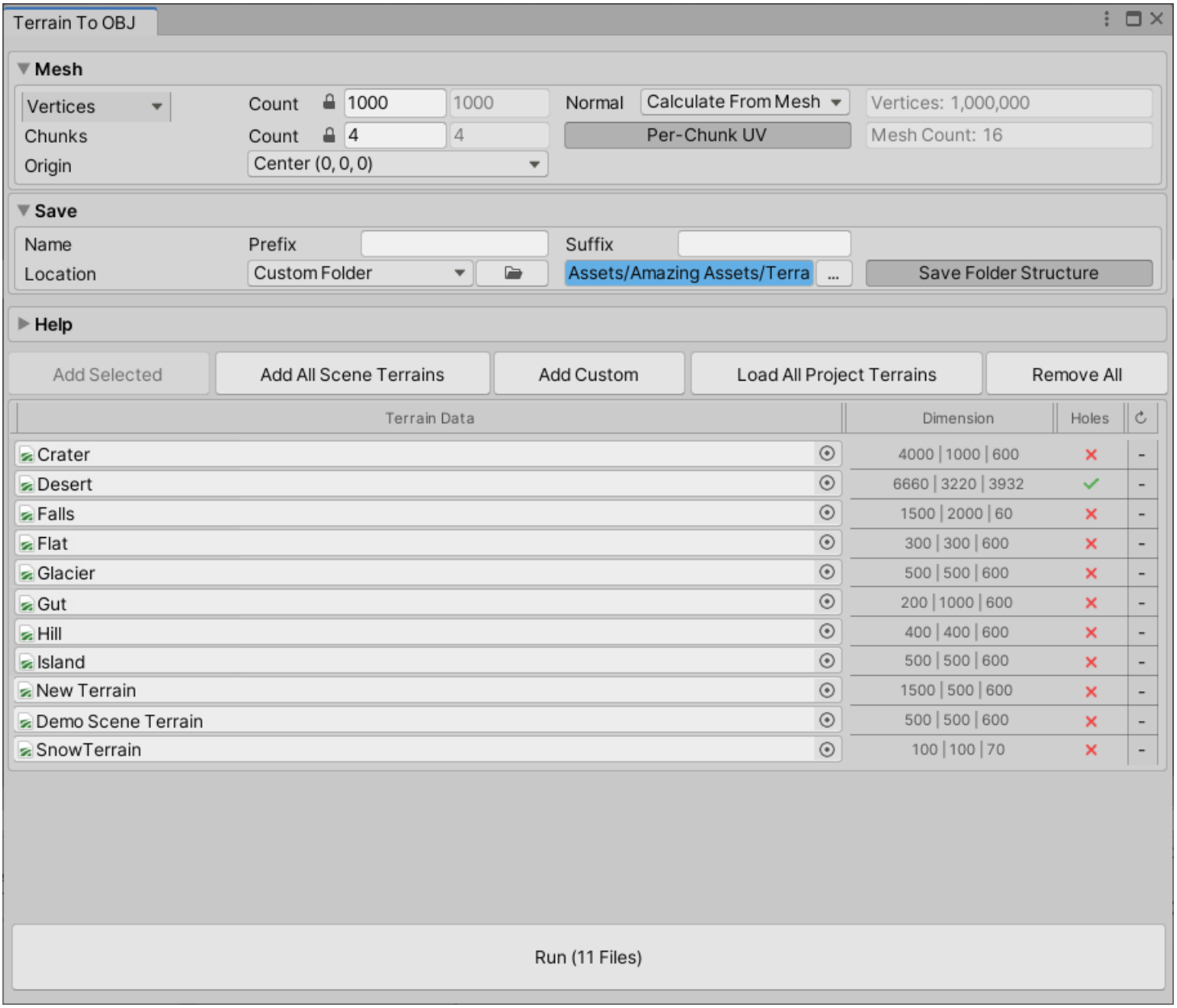
Terrain To OBJ

amazingassets.world

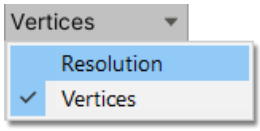
Editor Window

Terrain To OBJ editor window can be open from **Unity Main Menu -> Window -> Amazing Assets**. Target terrains can be selected using buttons available in the top side of the window or by simple drag & drop terrain objects from Hierarchy and Project windows, including entire folders with terrain data.

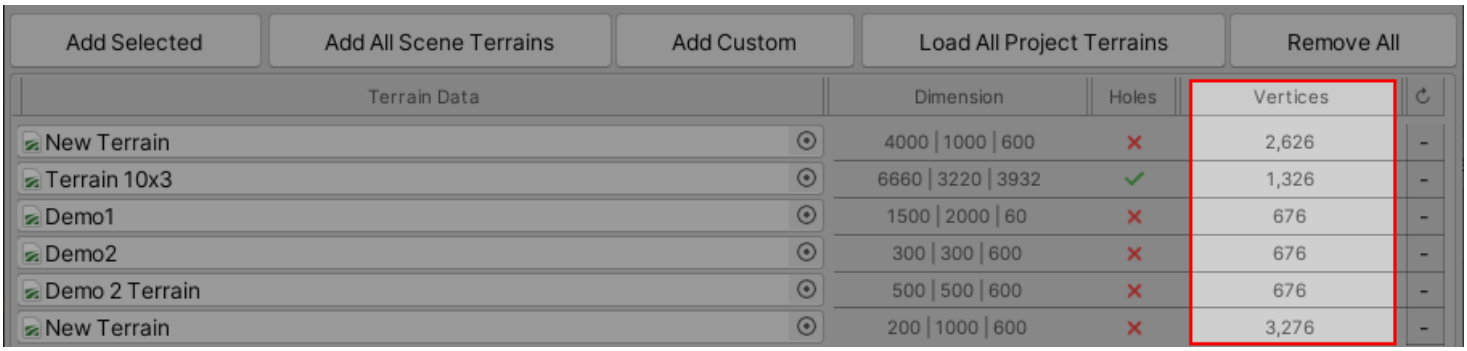


Vertices – Controls generated mesh vertex count horizontally and vertically:

Final vertex count is displayed in the upper right corner:



Choosing **Resolution** option instead of the **Vertices**, calculates vertex count horizontally and vertically in the way that vertex 2D grid always has quad shape. In this case final mesh vertex count depends on the source terrain length & width sizes and is displayed in the terrains list section.



Normals – Calculates and saves *Normals* for OBJ file. If generated OBJ file has no *Normals*, Unity (and any other 3d modeling software) calculates them manually after importing a mesh.
Reduces generated file size if is disabled, but may create visible *Normal* seams on the chunks edges.

Chunks – Splits source terrain into 2D grid and after that each part is converted into a mesh. Count property defines Horizontal and Vertical split amount.

Per Chunk UV – Defines UV layout for chunks. If enabled, all sub-meshes will have individual UV in the range of [0, 1]. Otherwise one UV layout in the range [0, 1] will be stretched over all chunks.

Origin– Generated mesh origin position can be in (x0, y0, z0) or the same as it has in a scene. If terrain does not exist in a scene and is selected from Project folder position (x0, y0, z0) is used.

File Name Prefix/Suffix – Add prefix and suffix for generated file. Useful when creating multiple OBJ file variations.

Save Location – Generated OBJ file can be saved in three locations:

- Same Folder – File is saved in the same folder as the source Terrain asset.
- Same Subfolder – File is saved in the same folder as Terrain asset, but inside a custom subfolder.
- Custom Folder – Allows to save OBJ file in any directory on the hard drive. Directory requires Read/Write permeations.

Run time API

Terrain To OBJ extension methods can be brought into scope with this using directive:

C#

```
using AmazingAssets.TerrainToOBJ;
```

Unity TerrainData class now will have two additional methods:

1. **ToOBJ** - returns **string** with OBJ file data.
2. **ToOBJStreamWriter** – same as **ToOBJ** method, but instead of returning sting data, calculated OBJ file is instantly saved into a file using **StreamWriter**, without allocating memory for string builder.

```
public string ToOBJ (int vertexCountHorizontal, int vertexCountVertical,
                     Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file as a single mesh.

public enum **Normal**

- None – Mesh has no normals buffer.
- CalculateFromMesh – Normals are calculated from generated mesh.
- CalculateFromTerrain – Normals are read directly from source terrain.

```
public string ToOBJ (int vertexCountHorizontal, int vertexCountVertical,
                     int chunkCountHorizontal, int chunkCountVertical,
                     bool perChunkUV, Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file as a collection of multiple sub-meshes (2D grid).

```
public string ToOBJ (int vertexCountHorizontal, int vertexCountVertical,
                     int chunkCountHorizontal, int chunkCountVertical,
                     int positionX, int positionY,
                     bool perChunkUV, Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file a collection of multiple sub-meshes (2D grid), but only segment defined by **positionX** and **positionY** variables is generated.

```
public void ToOBJStreamWriter (System.IO.StreamWriter streamWriter,
                               int vertexCountHorizontal, int vertexCountVertical,
                               Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file as a single mesh and using **streamWriter** is instantly saved into a file.

```
public void ToOBJStreamWriter (System.IO.StreamWriter streamWriter,
                    int vertexCountHorizontal, int vertexCountVertical,
                    int chunkCountHorizontal, int chunkCountVertical,
                    bool perChunkUV, Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file as a collection of multiple sub-meshes (2D grid) and using **streamWriter** is instantly saved into a file.

```
public void ToOBJStreamWriter (System.IO.StreamWriter streamWriter,
                    int vertexCountHorizontal, int vertexCountVertical,
                    int chunkCountHorizontal, int chunkCountVertical,
                    int positionX, int positionY,
                    bool perChunkUV, Normal normalReconstruction, Vector3 originPosition)
```

Terrain is converted into OBJ file a collection of multiple sub-meshes (2D grid), but only segment defined by **positionX** and **positionY** variables is generated and using **streamWriter** is instantly saved into a file.