

地形植被草使用说明

时间	修改内容	修订者
2019.12.9	初版 v1.0	吴俊

根据项目需求，对项目场景中的大地形植被草进行导出和渲染，移除植被草渲染对地形的依赖，这样场景中的地形、建筑和植被等完成大场景的工作流程的最后一环。然而实现的植被草相对 Unity 提供更好的灵活性，体现的优势如下：

- 1、提供植被草 GPU Instancing 和 Quad 合并两种渲染模式
- 2、根据相机和角色的范围，动态进行高效植被草的创建与渲染
- 3、可以使用第三方工具（如 PS）进行草的密度精确修改和人为定制
- 4、每个地形植被草的笔刷可以进行独立配置，能够实现不同的裁剪距离、显示密度、密度阈值、高度偏移、阴影显示、指定模型和材质

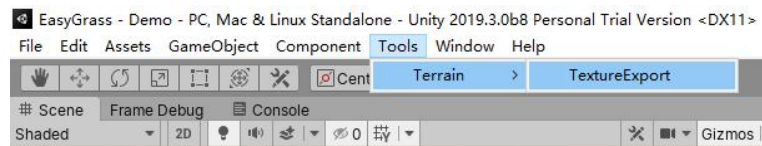
目前红宝石项目已经实现的效果如下：



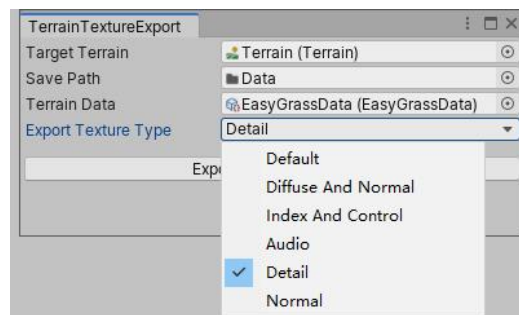
一、植被草数据的导出

为了解决已有的工作流程，复用之前已经使用的地形植被草，需要将地形植被草的数据导出，方便在进行植被草渲染时使用，解决对 Unity 地形的依赖问题，同时能利用 Unity 地形植被草的笔刷功能进行制作。

选中菜单栏 Tool->Terrain->TextureExport 选项



打开地形贴图导出窗口



Target Terrain:场景中包含 Terrain 组件要导出的地形

Save Path:保存地形各种贴图的路径

Terrain Data:保存地形配置数据，如果没有设置会在贴图保存路径下自动进行加载，无法加载则进行创建

Export Texture Type:导出贴图的类型，可以分类导出和同时导出所有，根据功能不同可分为：

1、Diffuse And Normal

地形漫反射和法线纹理数组，解决地形笔刷数量过多在 OpenGL ES3.0 的最大支持 16 层纹理限制

2、Index And Control

和 Diffuse And Normal 的优化目的相同，只是笔刷使用 Atlas 的方式存储，索引贴图存储笔刷的索引，控制贴图每一个通道存储笔刷的强度数据

3、Audio

地形声音数据，实现角色行走在沙地和草地上发出不同的声音。实际实现保存了地形笔刷混合最大层数，最大层数可以使用 Audio Layer Count 进行设置

4、Detail

地形植被密度数据，在植被草渲染时根据密度进行动态创建，密度越大草越密

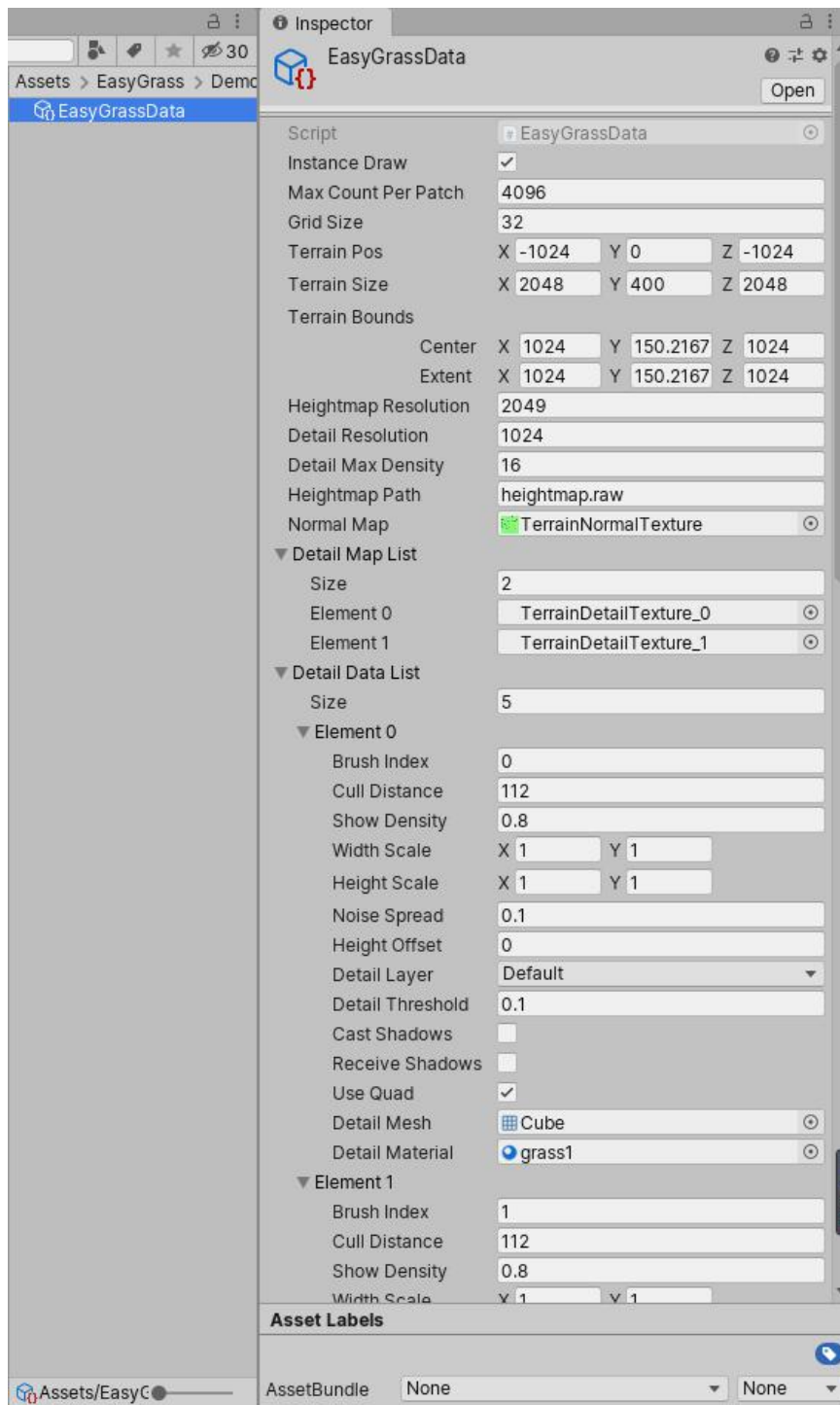
5、Normal

地形法线贴图，在植被草渲染时根据地形法线确定草的朝向

注意，这里我们只需要使用 4 和 5 的功能即可。

二、植被草数据的配置

导出之后的植被配置数据如下所示：



1、地形基础数据

Instance Draw:是否使用 GPU Instancing 进行渲染（细节说明见后面渲染部分）

Max Count Per Patch:在非 GPU Instancing 进行渲染时，同批次合并 Quad 的数量

Grid Size:地形被拆分网格的大小，根据项目需求进行设置和测试，太大和太小都会影响性能；

Terrain Pos:地形的世界坐标位置

Terrain Size:地形的宽度、长度和最大高度

Terrain Bounds:地形的包围盒

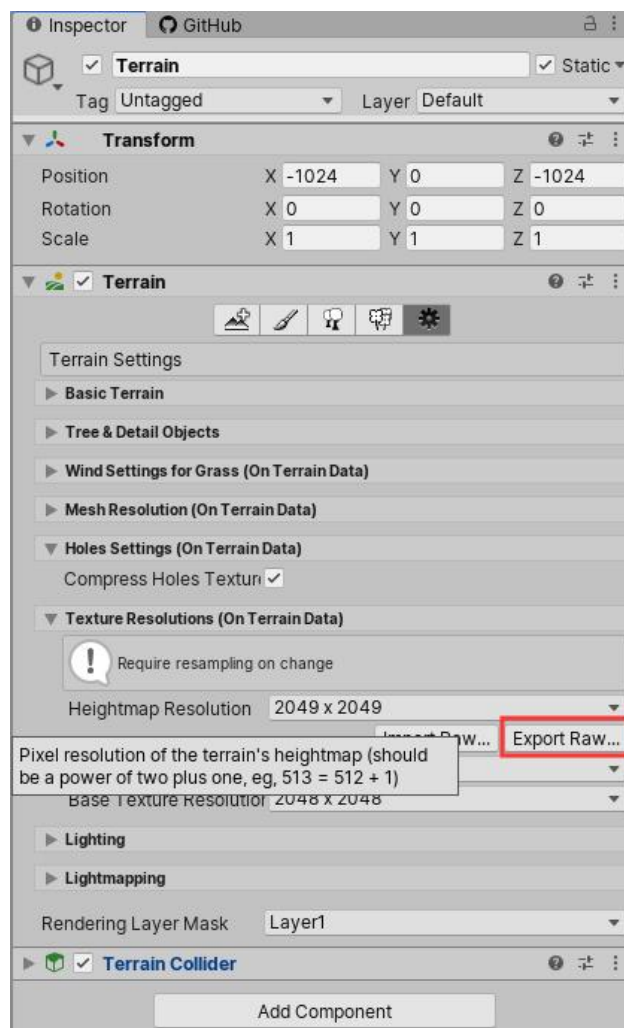
Heightmap Resolution:地形的高度图分辨率

Detail Resolution:地形的密度图分辨率

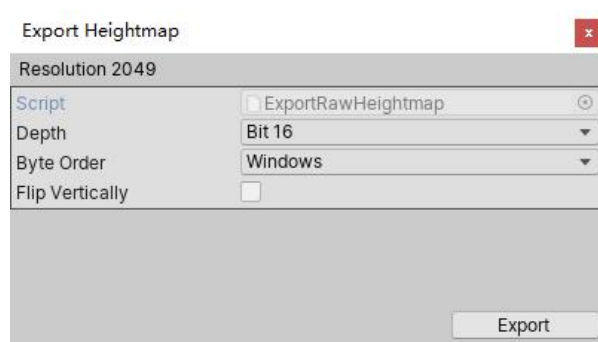
Detail Max Density:地形的最大密度

2、地形高度图数据

Heightmap Path: 高度图保存相对 StreamingAssets 的路径，注意必须保存在 StreamingAssets 和其子目录下，保存时选中包含 Terrain 组件的对象，在 Inspector 中打开 Terrain 组件的设置 Tab，选择“Export Raw...”按钮，如下图所示：



在显示的 Export Heightmap 窗口中进行如下设置：



Depth:导出高度图数据的精度，这里选择 **Bit16**

Byte Order:保存高度图数据的顺序，**Windows** 和 **Mac** 使用不同的大小端存储，这里选择 **Windows**

Flip Vertically:保存高度图数据是否进行垂直翻转，这里不要选中
最后点击 **Export** 按钮进行保存

3、地形法线数据

Normal Map:需要将地形植被工具导出的法线贴图设置到这里

4、地形密度贴图

Detail Map List:需要将地形植被工具导出的密度贴图设置到这里

5、地形笔刷数据

Detail Data List:地形植被导出工具保存的笔刷数据，数量与笔刷数量对应，笔刷的配置数据如下：

Brush Index: 地形笔刷索引

Cull Distance: 地形笔刷的裁剪距离

Show Density: 地形笔刷的显示密度

Width Scale: 地形笔刷宽度缩放范围，X 为最小比例，Y 为最大比例

Height Scale: 地形笔刷高度缩放范围，X 为最小比例，Y 为最大比例

Noise Spread: 地形笔刷创建植被影响噪声的因子

Height Offset: 地形笔刷创建时相对地形高度的偏移

Detail Layer: 地形笔刷渲染时使用的层

Detail Threshold: 地形笔刷创建时密度阈值，低于该密度不会创建

Cast Shadows: 地形笔刷渲染时是否投射阴影

Receive Shadows: 地形笔刷渲染时是否接受阴影

Use Quad: 地形笔刷是否使用 **Quad** 进行渲染，非 **GPU Instancing** 只能使用 **Quad** 模式

Detail Mesh: 地形笔刷 **GPU Instancing** 渲染可以使用 **Quad** 模式，也可以指定 **Mesh** 渲染

Detail Material: 笔刷使用的渲染材质，需要使用着色器“**Custom/EasyGrass**”，材质参数如下所示



Tint Color: 地形草的颜色

Main Texture: 地形草的贴图

Alpha Clip: 地形草的 AlphaTest 阈值

Show Range: 地形草的显示范围，用于远处草消失的过渡

Wave Speed: 草动画波浪速度

Wave Amp: 草动画波浪强度

Height Cutoff: 草动画的高度裁剪，确保草的底部不会移动

Wind Speed: 草动画的风速度

Wind Texture: 草动画的风的变化范围

Height Factor: 草动画的高度因子

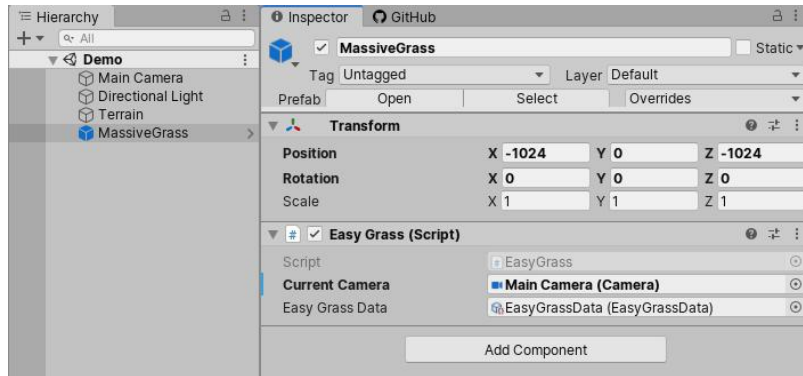
Terrain Position: 地形世界空间的位置

Terrain Size: 地形包围盒的大小

Enable GPU Instancing: 是否开启 GPU Instancing 渲染，注意不要在这里设置，需要设置配置数据的 Instance Draw

三、植被草的使用

使用时非常简单的，关键在于前期植被草的数据导出和配置。在使用之前需要将植被草的组件 `EasyGrass_v1.0.unitypackage` 导入到 Unity，导入之后在场景中新建一个 `GameObject`，然后添加 `EasyGrass` 组件，如下图所示：



Current Camera: 场景中当前使用的渲染相机

Easy Grass Data: 之前地形插件导出的植被草数据

四、植被草后期计划

- 1、对植被草的构建使用 Job System 和 Brust 进行多线程和汇编优化
- 2、优化现有数据结构减少运行时内存占用和 GC
- 3、优化植被草的着色器进行光照效果改善和 LOD 分级
- 4、支持植被草与对象（角色等）交互动画
- 5、实现植被草的踩踏、生长、四季、燃烧等效果

五、QA

1、GPU Instancing 和 Quad 合并如何进行选择？

GPU Instancing 在 Android 平台需要 OpenGL ES3.0，不保证 100%所有机型都能支持，并且 Instancing 在编辑器的最大数量 512 个，真机是 128 个，数量太少了，但是对于自定义少量大 Mesh 可以使用。非 Instancing 模式兼容性好渲染 DC 少，但内存占用较高，最好需要根据真机实测进行选择。

2、植被草笔刷数据如何配置才能获取最好性能和效果？

可以考虑减少笔刷数量、裁剪距离、使用 Quad、调整显示密度和密度阈值，都能减少渲染草的数量。