

Quick Introduction

Flooded grounds is an unity environment asset depicting gloomy, flooded rural area in post-apocalyptic/horror style. Comes as one large map as well as props, buildings, rocks and vegetation used to build it.

Performance

In order to estimate performance, it is recommended to check webplayer build (linked on asset page) on target hardware and refer to FPS counter (top left corner).

Please note that asset is not designed for mobile devices, so it will not perform optimally on them out of the box.

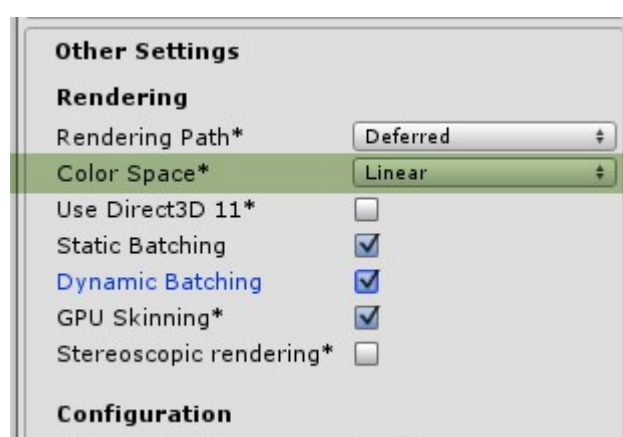
Known Issues

- EDITING TREES

Foliage in this project is created using Unity's tree creator system, which automatically generates texture atlases. Module that does that has some shortcomings in version of unity this project was published on (5.2.2f1) Namely, generated atlas resolution is only 1024x1024 pixels, textures come out dark in linear color mode and sometimes scene fog color gets baked down into atlases . You might have to disable scene fog and switch to gamma color space temporarily during editing process of trees.

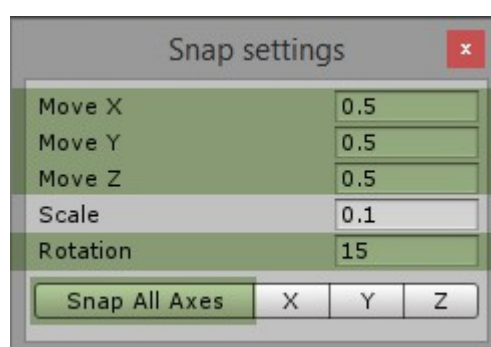
Project Settings

Asset relies on PBR shader introduced with Unity 5, which looks and works best in Linear color space. So, it is recommended to go to **Edit->Project Settings->Player** and switch color space from gamma to Linear. If your project uses forward rendering mode instead of deferred, you'll also need to select "Global Fog" component of Main Camera and disable "Distance fog" checkbox to avoid double fogging.

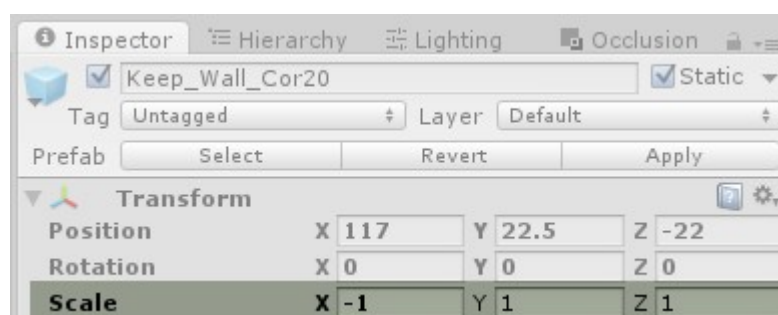


Grid And Snap Settings

Before using modular kits to assemble buildings, grid and snap settings have to be set first. Access it from **Edit->Snap settings** and set MoveX, MoveY and MoveZ steps to 0.5 units (50 cm). Use increment of 5 degrees for rotation.



If you want to mirror modules, you can simply use scaling value of -1 on respective axes.



Sample Scene and Preassembled buildings

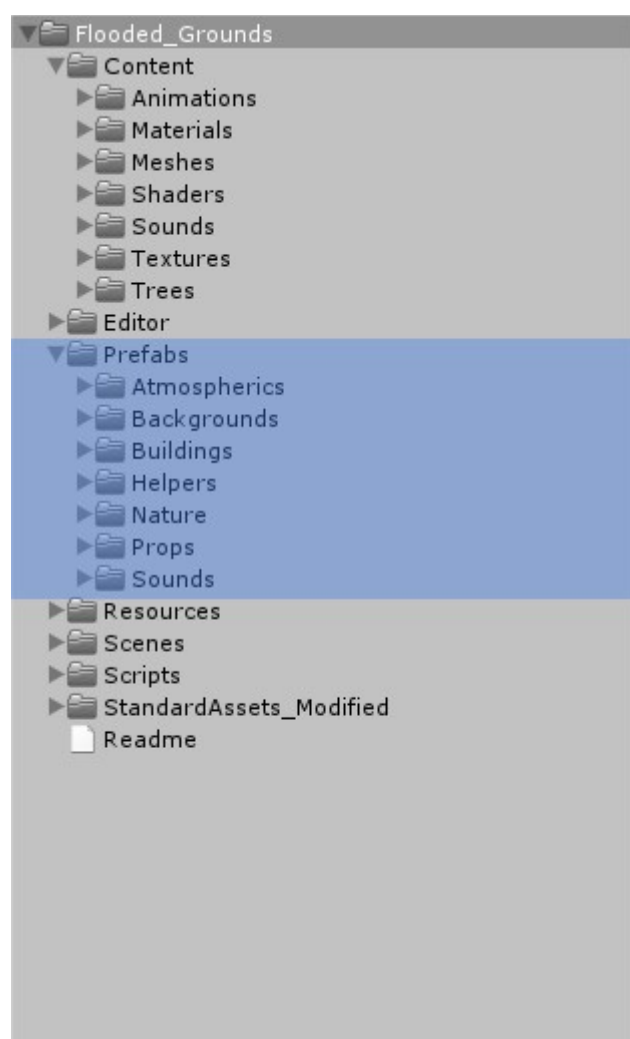
Open **/Flooded_Grounds/Scenes/Scene_A.unity** for main scene – checking this is the quickest way of finding about look, quality and style of assets as well as get sense how they can be used together in a cohesive manner.

/Flooded_Grounds/Scenes/PreAssembled_Buildings.unity contains preassembled buildings which should serve as a good quickstart. In order to use them in your scene either select respective parent object of building in hierarchy view and copy/paste it in your scene or make a prefab by dragging parent object from hierarchy to project view. Please note, that unity does not support nested prefabs, so if you decide to prefab whole building, prefab connections of individual modules will be lost.

Folder Organization and Structure

Folder structure of package is pretty self-explanatory. Most of the content you are looking for is in **/FloodedGrounds/Prefabs/** group and it's subfolders. Actual art content (meshes, textures, materials, animations, sounds) are in **/FloodedGrounds/Content/** folder. Subfolder breakdown of Prefabs folder is as follows:

- **Atmospherics** – various atmospheric effects – dust particles, falling leaves etc.
- **Backgrounds** – lower resolution assets that are used as backgrounds/objects that are only supposed to be seen from a distance.
- **Buildings** – contains buildings and their modular parts as well as various architectural structures
- **Helpers** – various helper objects, such as human figure for scale comparison purposes as well as collision blocker meshes
- **Nature** – contains various natural objects, such as bushes, trees, rocks, stones and grass.
- **Props** – various interior and exterior props
- **Sounds** – containing audio prefabs



Textures

/Flooded_Grounds/Content/Textures/ folder houses all textures used in this project. Textures are categorized by using prefixes. These are:

- **ATM_** - textures used for particles and various atmospheric effects
- **BGR_** - textures for background objects and/or skyboxes and reflection maps
- **BLD_** - textures for buildings
- **DET_** - detail textures, be it detail diffuse or detail normal maps for use with standard shader or as blend/overlay maps
- **GR_** - tiling textures and normal maps for terrain
- **NAT_** - textures for natural formations (foliage, rocks etc)
- **PROP_** - textures for props
- **MISC_** - misc helper and debug textures, such as grids, checkers etc.

Texture suffixes indicate actual type of texture map. For example:

- **TextureName_A** - Albedo/Diffuse map (RGB)
- **TextureName_AO** - Ambient Occlusion map.
- **TextureName_MS** - Metalness (R) and smoothness (A) map. Objects that are metallic are marked as white in R channel. Smoothness determines how smooth objects are (white = polished, black = rough)
- **TextureName_N** - Normal map.
- **TextureName_AS** - Albedo (RGB) and Smoothness (A) map. Used mainly as terrain textures.



Material/Shader

Asset mainly relies on Unity's standard PBR shader (Metallic setup). However, couple of objects use */Flooded_Grounds/PBR_TopBlend* shader that can blend tiling texture on top of up-facing portions of geometry. This can be moss, dirt, rust or any other kind of tiling texture.

Following is a detailed breakdown of material components:

- **Base Albedo (RGB)** – base albedo/diffuse map
- **Base Metalness (R) Smoothness (A)** – metalness/smoothness map for base. As in case of Unity's standard shader, metalness is stored in Red channel of RGBA texture and smoothness is stored in alpha channel.
- **Base Normal** – base normal map
- **Base AO** - base ambient occlusion map. This can be left blank if you do not have AO map.
- **Layer1 Albedo (RGB) , Smoothness (A)** – diffuse map for layer1 (blend layer). Use tiling textures for this slot. Blending happens on geometry that is facing upwards.
- **Layer1 Metalness** - Metalness slider for Layer1. It is mainly included for artistic control and should be left to zero (unless you are planning to blend tiling metal on top of existing material)
- **Layer1 Normal** – normal map for layer1
- **Layer1 Breakup (R)** – breakup texture for layer1 used to break up uniform look of blend. Shader uses only Red channel of this texture.
- **Layer1 Breakup Amount** – Slider for controlling breakup texture contribution to blend effect.
- **Layer1 Tiling** – Tiling multiplier for Layer1. Higher values mean more tiling
- **Layer1 Blend Amount, Layer1 Blend Height** – Use these two inputs together to fine tune how layer1 blends with base texture. Blend height can take both positive and negative values
- **Detail Normal** – detail normal map. This can be used to add fine grain on top of existing normal maps to create illusion of detail
- **DetailNormal Intensity, DetailNormal Tiling** – tiling and intensity controls for detail normal map

