Project "Increased Terrain Distance"

Author:

Michael Rauter (a.k.a. _Nystul_ from reddit)

Version:

1.0

Project Description:

This project aims to increase terrain/view distance of the Daggerfall Tools for Unity

Daggerfall Tools For Unity

Design Philosophy:

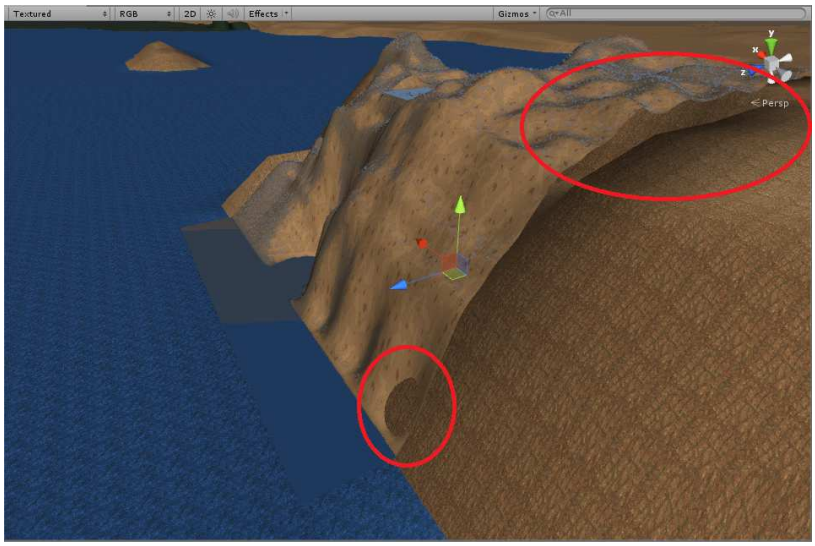
A world terrain object (using unity terrain) built from the world height map is maintained.
One main objective was to make everything work inside one script (and a shader for texturing) without a need for changes in any other files.
This means as a consequence that initialization of resources of other scripts that are used inside the *IncreasedTerrainDistance* script needs to be finished before any other work can be done.
Another objective was to minimize performance impact arising from loading and rendering the world terrain. I did some investigations and decided to use one big unity terrain object for the whole world. Experiments showed that this approach had the lowest impact on performance. The world terrain has to be created only once and only needs to be translated to match with the terrain geometry of the StreamingWorld component.
A consequence is though that low-detailed terrain geometry is also created at the same position of the detailed terrain inside the distance from the player defined by TerrainDistance.
This may result in intersecting geometry of the (low-detail) world terrain and the geometry of the terrain within TerrainDistance of the StreamingWorld component).
To decrease the chance of intersecting geometry world terrain heights are decreased inside a TerrainDistance radius (this area is called sink area in the script) and the world map is further translated down on the y-axis a bit. Both measures do not solve the problem completely (see Known Issues section).

How to set-up:

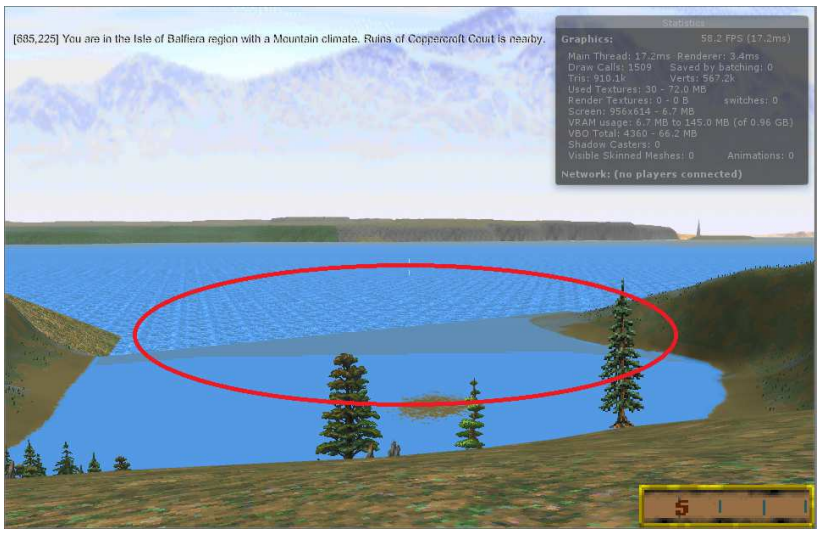
- copy script *IncreasedTerrainDistance.cs* into Scripts/Terrain folder in DFTFU project structure
- copy shader file *DaggerfallIncreasedTerrainTilemap.shader* into Shaders folder in DFTFU project structure
- Now there are 2 possible ways to integrate the *IncreasedTerrainDistance.cs* script into your project:
    - Add as script to StreamingWorld Component:
      Select Exterior/StreamingWorld Component,
      click Add Component->Scripts->Daggerfall Workshop->Increased Terrain Distance
    - Add as script to new GameObject:
      Create empty GameObject in Exterior, reset transform,
      click Add Component->Scripts->Daggerfall Workshop->Increased Terrain Distance
- Make sure Camera in PlayerAdvanced is tagged as MainCamera
- Create 2nd Camera-GameObject and rename it to something like secondaryCamera
- Select IncreasedTerrainScript in Unity Inspector and drag-in or select StreamingWorld instance, Player GPS instance and Secondary Camera (the one that was created before)
- Drag it into PlayerAdvanced and delete Flare Layer, GUILayer and Audio Listener in Unity Inspector
- Copy transform values from camera to secondaryCamera ("copy component" on transform of camera and "paste component values" onto secondaryCamera transform) – make sure values are identical
- Add script Player Mouse Look to secondaryCamera and clone all values from Player Mouse Look component in Camera – make sure values are identical
- Open menu point Edit->Project Settings->Tags and Layers and choose an empty layer and insert the exact name "WorldTerrain"
- Select PlayerAdvanced/Camera and put in these values:
    - Culling Mask: Everything except WorldTerrain
    - Near clip distance: 0.07, far clip distance: 5000
- Select PlayerAdvanced/secondaryCamera and put in these values:
    - Culling Mask: only WorldTerrain
    - Near clip distance: 1600, far clip distance: 800000
- Open menu point Edit->Render Settings and put in these values:
    - Fog Mode: exponential
    - Fog Density: 0.000005 (you can play around with this while in game mode later)
- Set TerrainDistance in Script StreamingWorld in Unity Inspector to a value higher than default, I suggest 4 (otherwise you will hardly ever see much of the world terrain)

Known issues:

- secondary camera is needed to decrease/prevent floating-point precision issues
- Player Mouse Look script values can get out of sync – need to rethink a better way to let the 2 cameras look into the same direction
- world terrain can still intersect with detailed terrain (the terrain inside TerrainDistance of the StreamingWorld script) – see the small red circle,world terrain and detailed terrain do not fit together (due to the way the unity terrain component is used) – large red circle:



- sometimes it happens that the player falls through the world, e.g. fast-travel to Wayrest (by pressing “2” on the keyboard) and depending on where you are put exactly you may fall through the world. I have to investigate this issue – if it is a side-effect of my script I have to find a solution
- there is much room for improving texturing of world terrain
- currently no seasonal texture support
- I don’t know what causes this issue:



- Daggerfall’s height map is boring flat (except shore lines which are way too steep) and has further lots of errors one can spot