10/02/2018 BLOG – Fog of war system used in XCON – Making it in Unity. C++ Fog of war

Welcome to the second update to be posted. This week I spent my time looking into fog of war implementation within Unity and networking.

Advance group project:

-Fog of war system

-Why use it?

For the stealth game which I am creating with my peers we wanted a fog of war system. Why would that be? Well due to the game being a stealth game we wanted the players to not know where anyone was. Luckily there are a wide range of implantations which can achieve fog of war.

-Implantations used Plane above level (RTS) / Frag shader / Projector?

One implementation which I am sure anyone would know is using a plane above the level and using a shader to mask off areas to show the visible and unviable parts of the map. This is a cheap and easy way to create a fog of war effect.

Another method is to change what the fragment shader colour outputs. This was a method which I didn’t know you could do until I ask one of my lectures at university who showed it to me. This method produces an effect like the one in the XCOM series [image(xcom\_fogOfWar\_1.jpg, <https://www.geforce.com/games-applications/pc-games/xcom-enemy-unknown>] which was the effect we wanted.

Here is a similar effect achieved in C++ with OpenGL (More next week).

[Gif opengl]

The last method to discuss here is using a projector component within Unity. The projector component allows you to project a material onto all object that intersects with its frustum (<https://docs.unity3d.com/Manual/class-Projector.html>). Using a projector would allow you to project a material which makes everything in your scene ‘hidden’. Then you can blend a fog of war render texture and mask areas which should be visible and not visible. [Video, <https://www.youtube.com/watch?v=lojB-G9xkCc>]

-HDRP issues found?

Whilst all the above methods work there is one small issue with the project. That small issue is that we are using the HDRP (High Definition Render Pipeline). Before I go any further, I should explain that the HDRP is amazing. The visual quality that it gives to Unity is something you must see for yourself [Blog, <https://blogs.unity3d.com/2018/09/24/the-high-definition-render-pipeline-getting-started-guide-for-artists/>]. And while there are issues for my certain case which I will go though the HDRP is a good step in the right direction.

The issue which I have run into obvious involves how Unity renders its scene. If there was no HDRP in the project, then a simple replacement shader could do the job. Not very well mind you as lighting would have to be done manual but it could be done. Another option and the one that I prefer which is stated above. And that is using a projector to project a material in the scene then use render textures and blend then together to get the desired effect.

From what I have tried. The HDRP does not allow you to use a replacement shader. This is understandable as if using the HDRP you want the benefits of it. And using a replacement shader would defeat the purpose of using the HDRP. However, projector also seems to not work for what ever reason. This could be down to the HDRP still being in preview.

-Alterative ideas?

So, I can’t at the moment figure out how to create a fog of war system like XCOM has. That’s fine. Game development is all about trying new things, find barriers and solutions to them. So, in the meantime I have contacted one of Unity evangelist in hopes of some help.

Not all hope is lost. As there are alterative ideas to the fog of war system. One being using a collider of sorts and checking if another player is within the collider then allow them to become visible. This could be done by turning on their mesh renderer or using a dissolve shader. Another idea could be to render any room you are not in with a solid colour. This would not allow the player to see what was in the other room. Effectively doing the job of the fog of war system.

\* If anyone has any tips or ideas please contact me. I would love to hear them. \*

Not only have I spent time looking into a fog of war system, but I have also setup a network within Unity.

-Networking

-Setup?

The network system which was setup is using Unity the Multiplayer High Level API (HLAPI). -Works?

[gif\_1.gif]

As shown in the gif I am able to connect clients to a server and sync movement. There are improvements to be made. The main one being reducing the noticeable latency between the clients. While the gif doesn’t show it there is a different on the two instances. The instance which is acting as the host and has a client has no lag while the solo client has lag from the host/client.

In the next few days I hope to reduce this and how smooth movement from both instances

-Improvements?