Meraki 1.o [doc 2]

Sahaj Garg
Masters of Animation, Games
and Interactivity (MAGI),
School of Design,
RMIT University,
Melbourne, Australia
s3738367@student.rmit.edu.au

Abstract

Meraki is a concept exploring unique 360 environment interaction to reflect on inner and outer space.

It explores some unique camera dynamics inside through VR. 3D audio and how to combine technoligies alike for some unique experiences, possibly questioning perceptions.

This document is also a follow from a previous MERAKI document.

Authors Keywords

Unity ; Optimisation ; Starsystems ; VR ; WebVR ; Scripting ;HTCVive ; Oculus GO ; Graphics ; Shaders ; lostincelestial

WEEK 7 Initialisation ~
Extending on my previous query of how the final idea should show, as in form of app, webpage or vr standalone.

~Time on researching : you don't even know what you don't know.

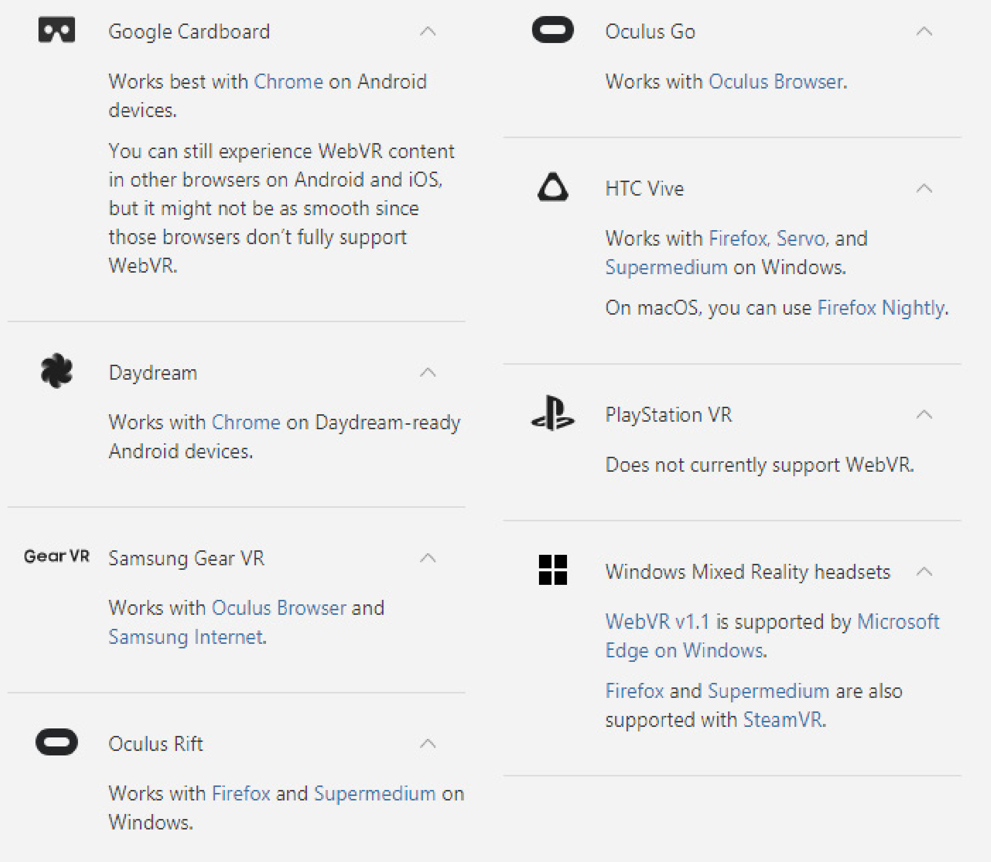
My intention is to reach maximum audience with most ease of access. Researching more on the same I came across WEBVR API.

WebVR is OpenGL output as webpage in which user has ability to switch to VR if a headset is connected or browse through normally if headset is not connected on a webpage.

To test out the project capabilities through WebVR potentially. I've broken it into smaller tests and initial hurdles I'm looking at which are as follows:

- **Camera Tests :** (only targeted camera for firsts tests)
- **Render/Graphic Quality tests**: Trying to anwer the query 'can we get reflections/refractions through webvr api. to what quality?)
- File Heaviness / Polycount
- Animations : Can webvr support dynamic animations? baking techniques , animation trigger and more.

(represtented on project map)



Google Cardboard

Works best with Chrome on Android devices.

You can still experience WebVR content in other browsers on Android and iOS, but it might not be as smooth since those browsers don't fully support WebVR.

Daydream

Works with Chrome on Daydream-ready Android devices.

Samsung Gear VR

Works with Oculus Browser and Samsung Internet.

Oculus Rift

Works with Firefox and Supermedium on Windows.

Oculus Go

Works with Oculus Browser.

HTC Vive

Works with Firefox, Servo, and Supermedium on Windows.

On macOS, you can use Firefox Nightly.

PlayStation VR

Does not currently support WebVR.

Windows Mixed Reality headsets

WebVR v1.1 is supported by Microsoft Edge on Windows.

Firefox and Supermedium are also supported with SteamVR.

Extract from WebVR info website on supported mediums.

Initiations: First WebVr tests / Understanding and setting up the project.

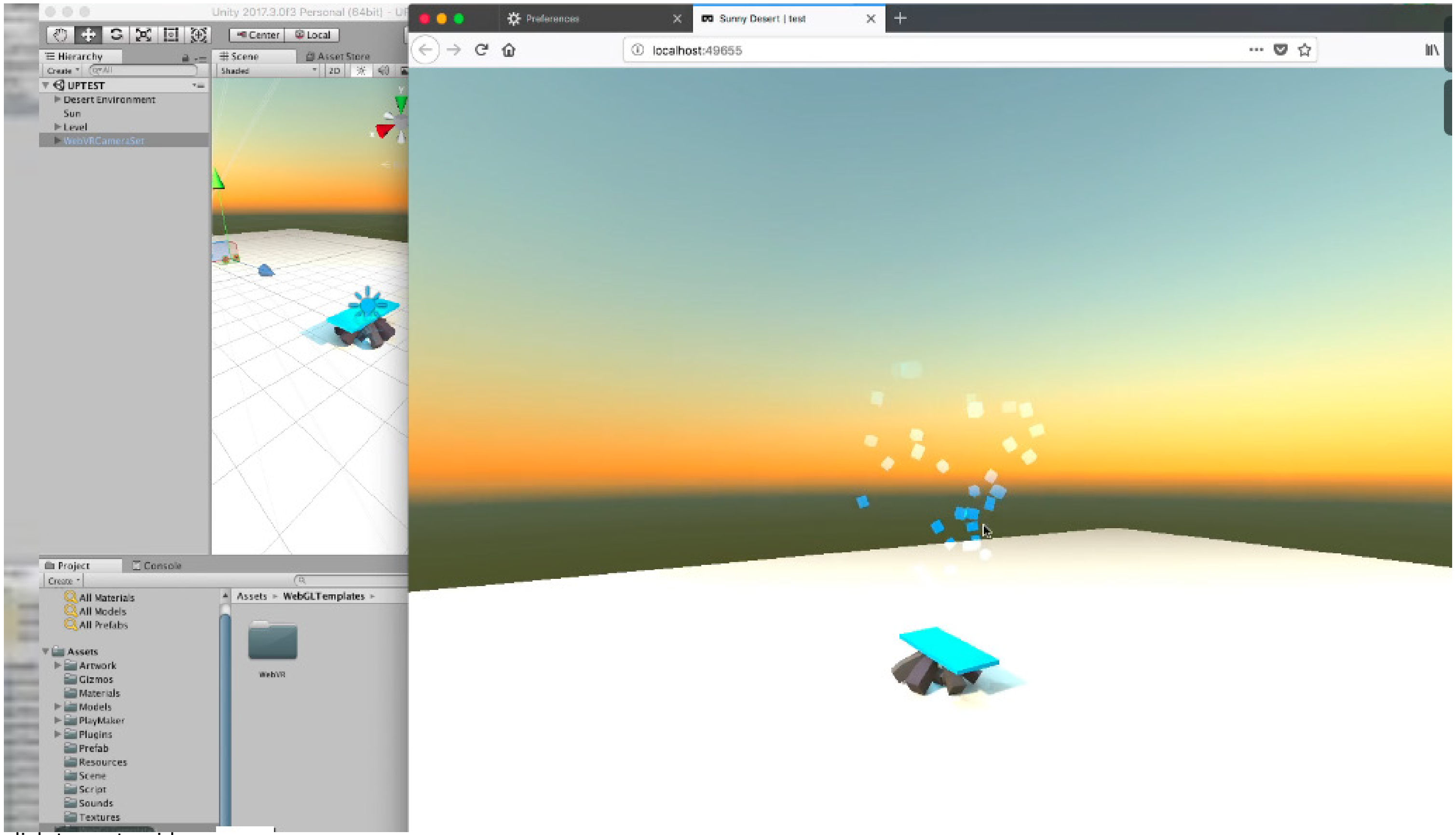
edit : I also looked at playmaker in unity and from the class demo’s though I just sticked to scripting for the project as I realised it gave more felxibility and being familiar with little coding it was easier.

WEBVR test : working demo , (with normal camera movements and mouse controls). (not in VR) [click image to go to video]

UNREAL Engine: Previous Hotseat seat question and which keeps bumping up during the project.

As my previous prototype was in UNREAL, I initially looked to integrate WEBVR Api to unreal, as of 2 sept,2018 , there’s not much information on this subject and succesfull webvr unreal integration.

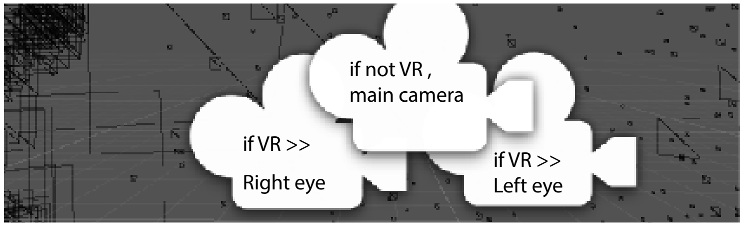
I proceeded with Unity, as It’ll save some time, instead of trying to figure out a way to understand Unreal with WEbvr integration. There are also certain factors as unreal and webvr api use



click to go to video.

UNDERSTAND WebVR CAMERAS . Developing Scripts

Here I studied how Camera setup for VR works.

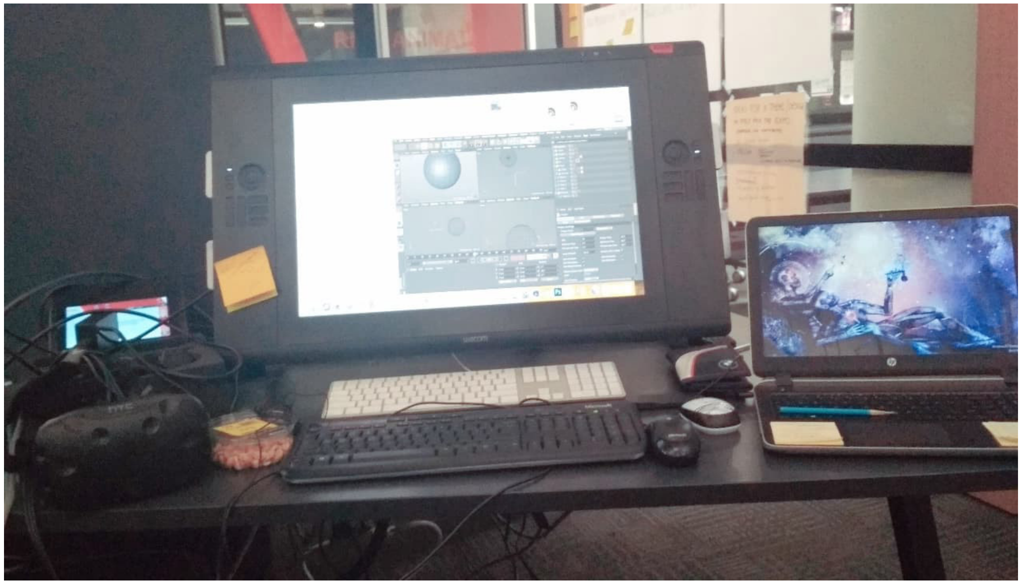


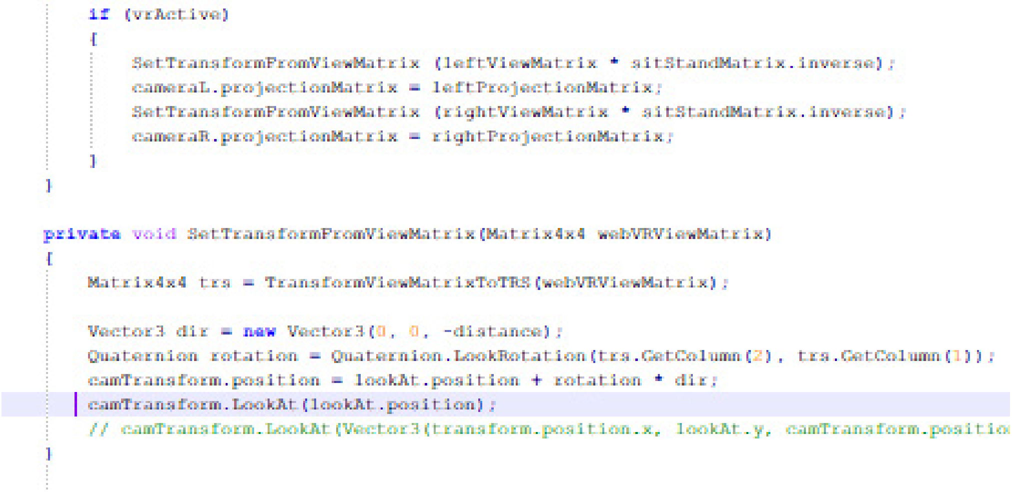
[switching to dual camera (left and right eye) if vr is enabled. // provided in WebVR sample scene]
I further needed to **take up gyro / headrotation data from VR through WebVR API and apply it to camera transformation.**

One of the Ideas I looked and researched upon was also Rotating Object(/scene) from headset data instead of changing camera position from headset data, while keeping the camera in fixed place. I guess this will make navigating the camera through a scene easier.
EDIT: though I'm approaching the other method and rely on switching scenes to change camera position, as in this scene it'll be locked through VR gyro Data and script.

Getting it work in normal view- on screen was quick but everytime I swichted to VR and tested it out , it didn't work. It was eathier cross eyed, stuck or overrided it to normal camera dynamic, also I realised I need to restart pc after every test to reset configuration from previous tests. - yes i tried deleting temp and cookies(didnt work).

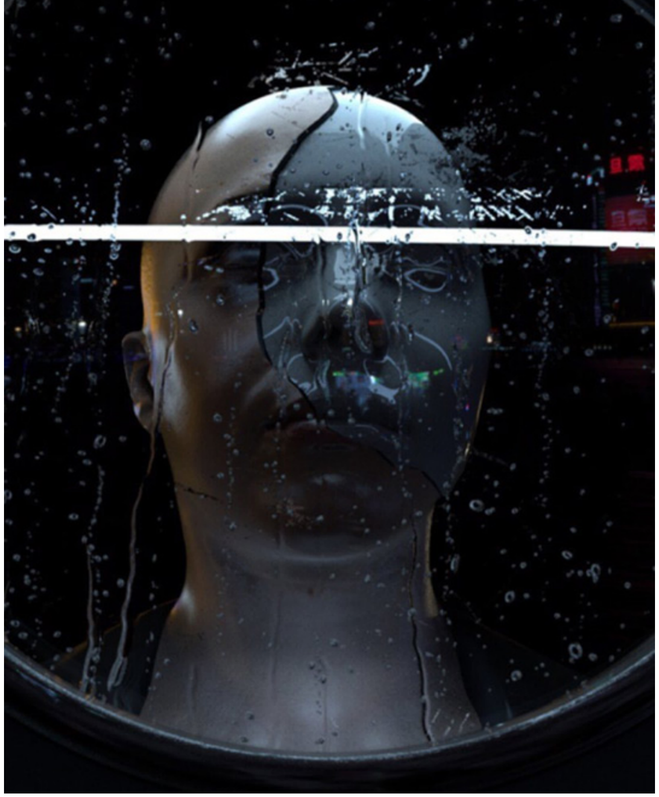
I started to study and develop the codes and webvr api language.
After a good amount and days of '¿happy frustration' I was able to get the lookat camera dynamic working with VR.

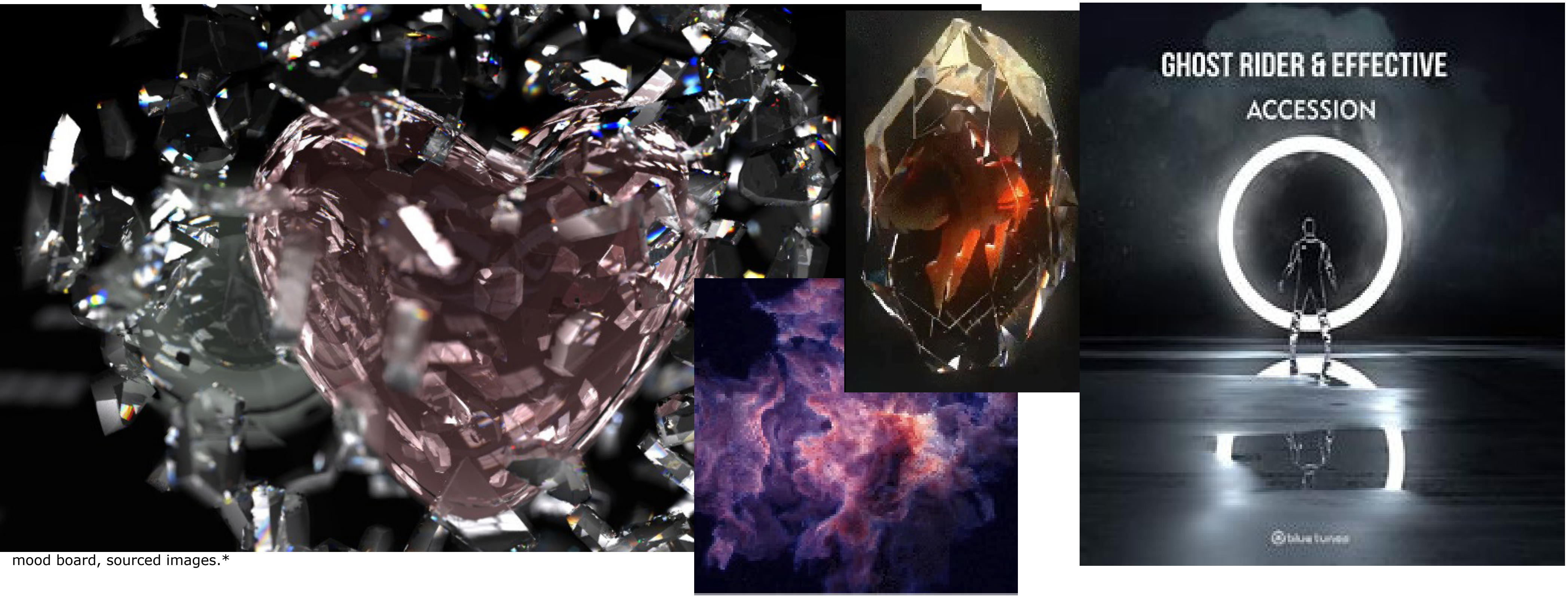




```
    if (vrActive)
    {
        SetTransformFromViewMatrix (leftViewMatrix * sitStandMatrix.inverse);
        cameraL.projectionMatrix = leftProjectionMatrix;
        SetTransformFromViewMatrix (rightViewMatrix * sitStandMatrix.inverse);
        cameraR.projectionMatrix = rightProjectionMatrix;
    }
}

private void SetTransformFromViewMatrix(Matrix4x4 webVRViewMatrix)
{
    Matrix4x4 trs = TransformViewMatrixToTRS(webVRViewMatrix);

    Vector3 dir = new Vector3(0, 0, -distance);
    Quaternion rotation = Quaternion.LookRotation(trs.GetColumn(2), trs.GetColumn(1));
    camTransform.position = lookAt.position + rotation * dir;
    camTransform.LookAt(lookAt.position);
    // camTransform.LookAt(Vector3(transform.position.x, lookAt.y, camTransform.positio
}
```

I now knew logic behind projecting on left and right eye individually if VR and developed lookat targetted script from gyro of headset. I also looked upon matrixes, quaderanions and more and learn't a lot of unity coding (javascript) of which I had little knowledge and haven't touched on for a while. ~ webvr api.





mood board, sourced images.*

Render Texture / Graphic Quality.
Refractions and Refractions// Realtime.

Unreal : - I again thought of unreal engine here as it shows good graphics straight out of box, but I realised they have effects loaded by default and in unity one needs to do and tweak that. Hence Unity

It's really interesting and a little annoying that how I've started to notice reflection's and refractions more around in life. I just can't help but uncounciouslly study how light is playing out and how it could be achieved in cg.

generating stuff with raytracing,loads of reflections and more in 3D - which takes good enough of time to render,now working with Unity which needs to render graphics in realtime , to achieve the the quality I have in mind is not as straight forward as thought to be. Along the lines of research to improve graphics in Unity these are the terms I looked into:
keywords/?/:

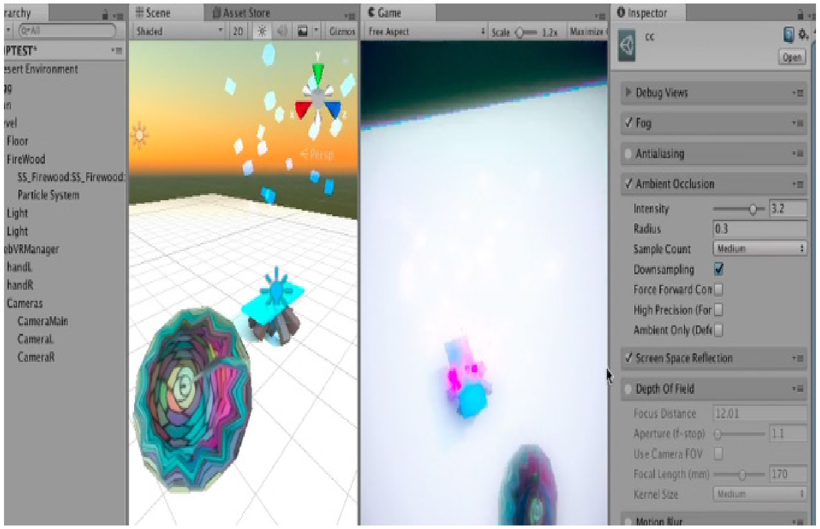
Post Processing - Effects rack by Unity.
Screen Space Reflection (Part of Post Processing)
HDRP - High Defination Render Pipeline
The Lightweight Render Pipeline
Cube Maps
Reflection map.

After Presentations, Max suggest to look more upon what the experience is . This has shifted my direction again and for next weeks I'm reflecting on the same where earlier I planned to look only in achieving the shader quality.

Sometimes I just put on a peice of music and start throwing strokes around. just a log of random brain fog.

these just help me unwind.

i need to declutter~.



▶ Debug Views
✓ Fog
○ Antialiasing
✓ Ambient Occlusion
Intensity 3.2
Radius 0.3
Sample Count Medium
Downsampling ☑
Force Forward Con □
High Precision (For □
Ambient Only (Defe □
✓ Screen Space Reflection
○ Depth Of Field
Focus Distance 12.01
Aperture (f-stop) 1.1
Use Camera FOV □
Focal Length (mm) 170
Kernel Size Medium
○ Motion Blur

Playing around with post processing in unity. Camera effects.

Brainfog1.0

User Feedback and research on Camera Dynamics:
with, by Nick and Evan

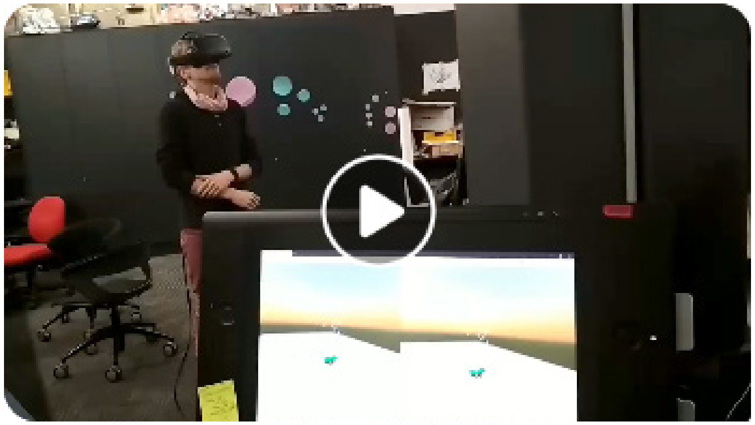
The vertical 'Y' axis is breaking the immersion, or feels less likely.
What if we invert the axis's?
what if we can keep y axis normal and x targetted?

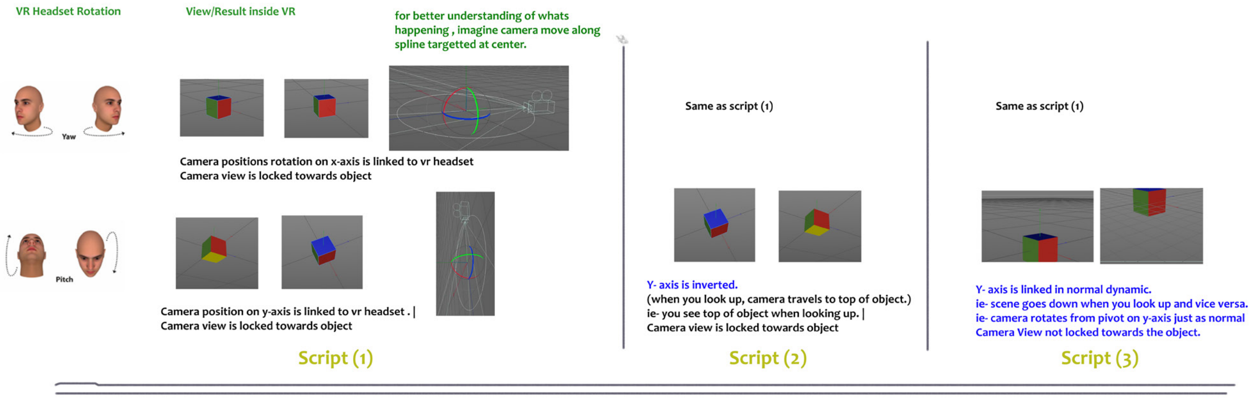
Different ideas for alternate camera scripts: (until now)

- Script(1) : [Targetted third person] Here camera is targetted towards a central object and revolves around it, which is linked to head rotation of VR headset - on all axis.(like a toroid - inside outside dynamic)

- Script(2) : [Targetted third person inverted y] The camera dynamic is same as 1, only difference being inverting the Y axis. Prototype testing revealed this subtle change can impact perception immensly.

Script(3) : [Hybrid , semi targetted] In this camera dynamic, camera revolves around targetted object(center) at x-axis , while on y-axis it's normal dynamic .ie- camera rotates from pivot (only on Y-axis)





for better understanding of whats happening , imagine camera move along spline targetted at center.

Camera positions rotation on x-axis is linked to vr headset
Camera view is locked towards object

Camera position on y-axis is linked to vr headset . |
Camera view is locked towards object

Script (1)

Same as script (1)

Y- axis is inverted.
(when you look up, camera travels to top of object.)
ie- you see top of object when looking up. |
Camera view is locked towards object

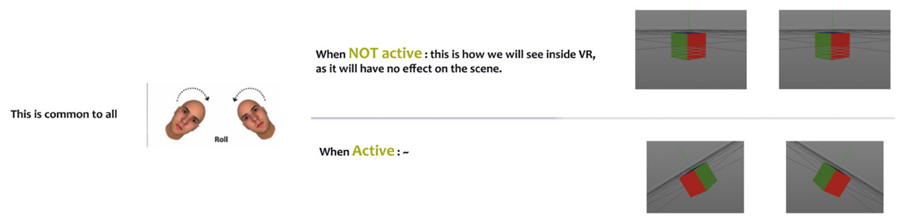
Script (2)

Same as script (1)

Y- axis is linked in normal dynamic.
ie- scene goes down when you look up and vice versa.
ie- camera rotates from pivot on y-axis just as normal
Camera View not locked towards the object.

Script (3)

Edit: At this moment , i was able to develop script (1) and (2) though for (3) needs some relogic and to write the whole script from scratch , which I'll like to focus on later.



This is common to all

When NOT active : this is how we will see inside VR, as it will have no effect on the scene.

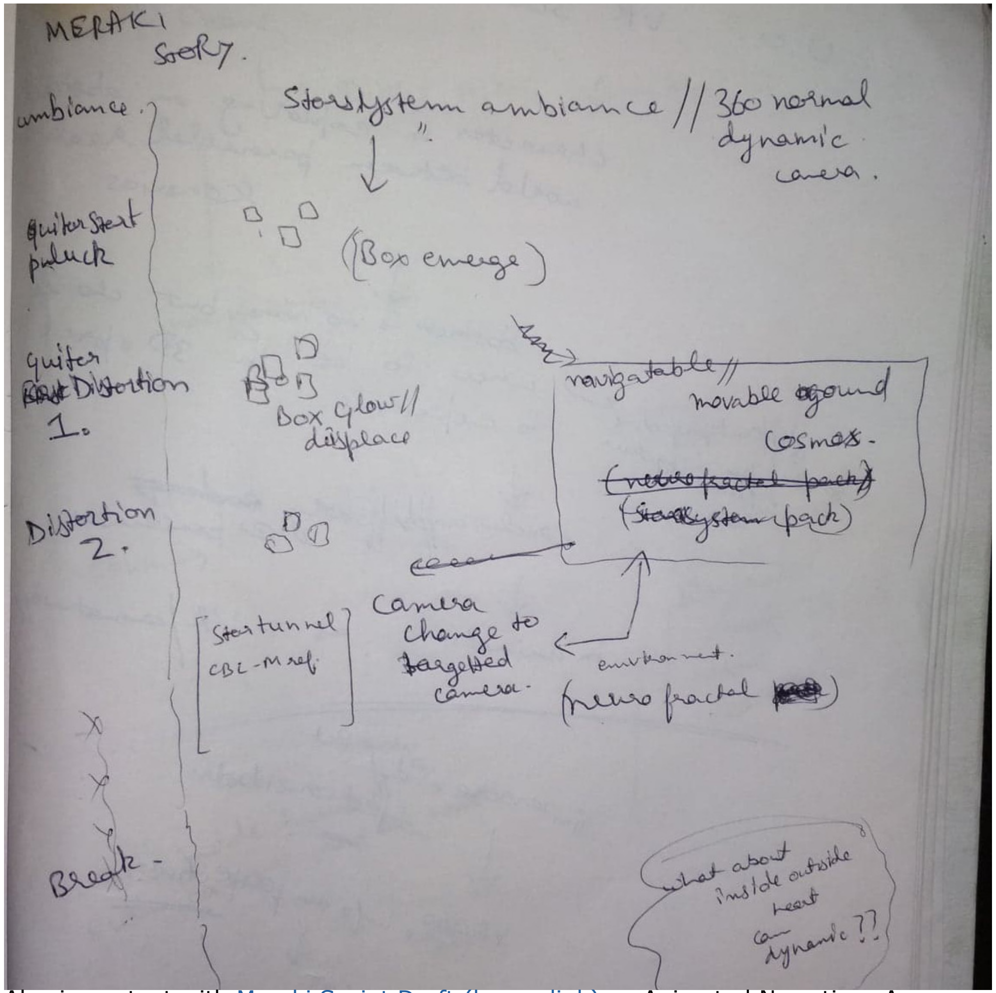
When Active : ~

I'll be developing a prototype using what has done been achieved till now, ie- script 1 or 2.

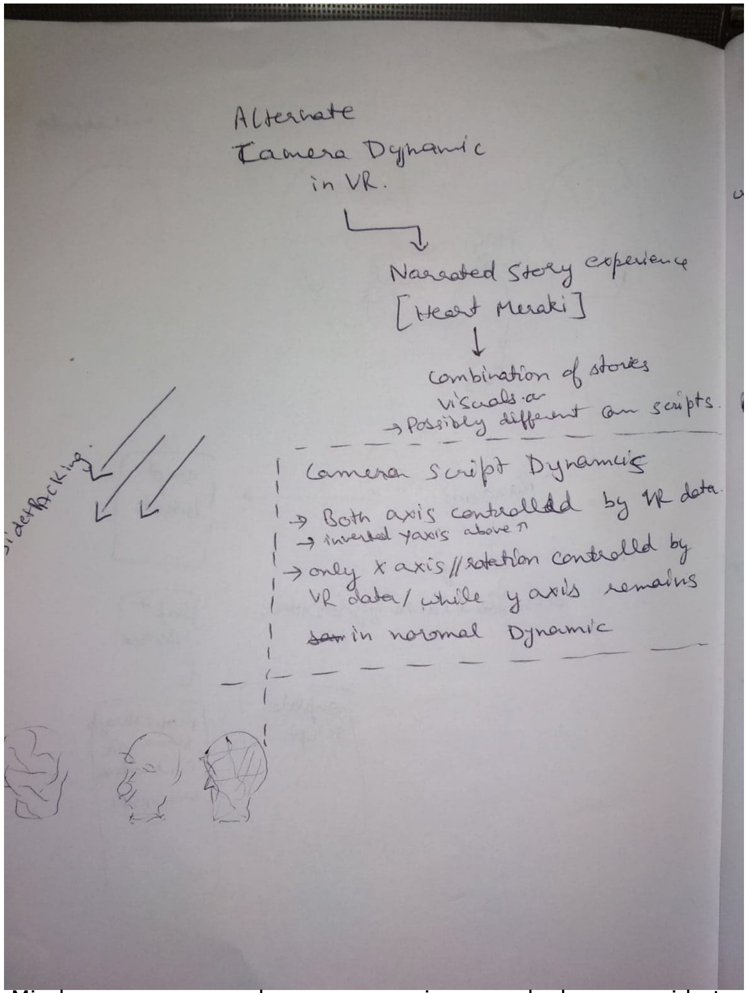
sz.Sahaj
s3738367@student.rmit.edu.au

During-after weekly presentation , I was planning to develop skills on graphic quality, on which reflections from John really helped on direction I got to look at (though kept for later right now) and Max suggests I should give a thought and reflect on what the experience is!

Doing so I realised I have a certain direction I'm aming for but the way is too foggy and here I'm reflecting on the final goal/vision/experience and more :


MERAKI STORY.
ambiance.
Starsystem ambiance // 360 normal dynamic camera.
Guitar Start pluck
(Box emerge)
Guitar ~~Beat~~ Distortion 1.
Box glow// displace
navigatable// movable ground cosmos.
~~(neuro fractal pack)~~
(~~Star~~System pack)
Distortion 2.
Star tunnel CBL-M ref.
Camera change to ~~targetted~~ camera.
environment.
(neuro fractal ~~pack~~)
Break -
what about inside outside heart cam dynamic ??

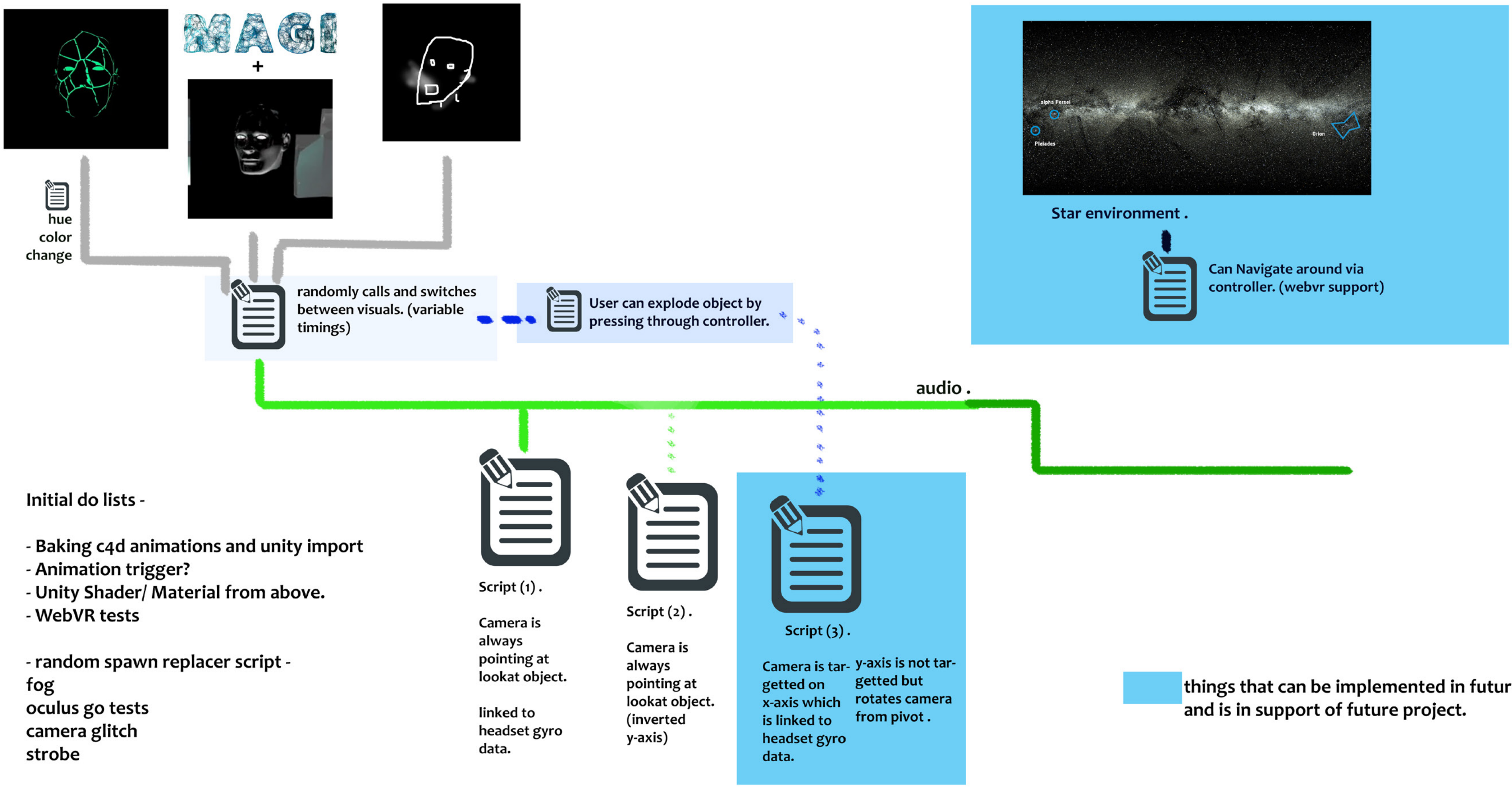
Also in context with [Meraki Sceipt Draft (hysperlink)](#) , - Animated Narratives A


Alternate Camera Dynamic in VR.
Narrated Story experience
[Heart Meraki]
combination of stories visuals or
→ Possibly different cam scripts.
Sidetracking
Camera Script Dynamics
→ Both axis controlled by VR data
→ inverted y axis above π
→ only x axis // rotation controlled by VR data / while y axis remains ~~same~~ in normal Dynamic

Mindmap on camera dynamcs,experience and where we side track.

I realised to achieve the final vision of an animated music experience , there are still lots of pre work/technologies which are needed to be developed and worked on. For the semester I decide to sidetrack the project into a simpler using tools which are developed till now and developing some which contributes to the finale - Meraki " which is now visualised as a interactive experience with a story, animation, multiple camera dynamics ,parallel scripts~etc.

SIDETRACKING !! ~
to the race with time for this sem-end.
Idea 1 : taking script(1) from camera dynamic, which was done at this point and combining it with some earlier visuals I did with some new ones in mind which will be called at random and more (in VR)-



hue color change

randomly calls and switches between visuals. (variable timings)

User can explode object by pressing through controller.

Star environment .

Can Navigate around via controller. (webvr support)

audio .

Initial do lists -

- Baking c4d animations and unity import
- Animation trigger?
- Unity Shader/ Material from above.
- WebVR tests

- random spawn replacer script -
fog
oculus go tests
camera glitch
strobe

Script (1) .

Camera is always pointing at lookat object.

linked to headset gyro data.

Script (2) .

Camera is always pointing at lookat object. (inverted y-axis)

Script (3) .

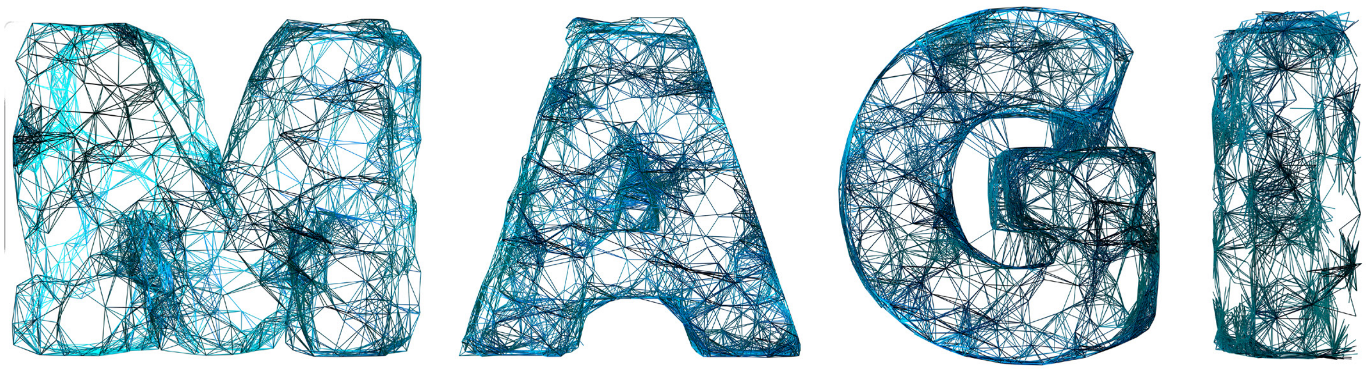
Camera is targetted on x-axis which is linked to headset gyro data.

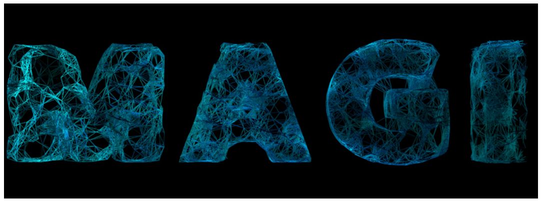
y-axis is not targetted but rotates camera from pivot .

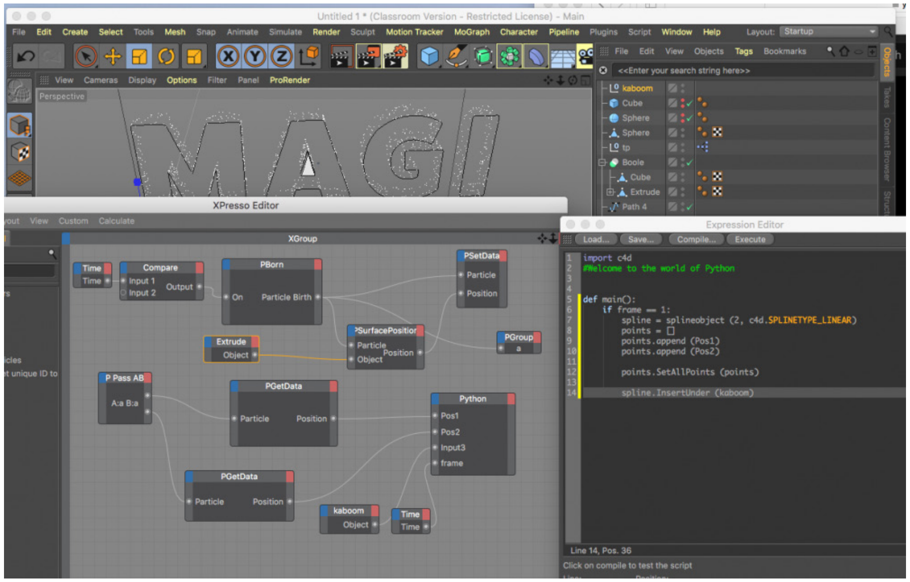
🟦 things that can be implemented in future . and is in support of future project.

All scrpits in support in and through WebVR API.

Reflecting on this for while I travelled back to the idea where Kate has asked”if we can look inside the heart” as well, which is real interesting. Now the idea developed into ‘script(1)Camera Dynamic(currently achieved 99%), ie-targetted on the outside of heart and a normal camera dynamic on the inside. It doesn’t followup on the meraki story ie-with full animation script but with just a visual and a much deeper experience which is in context with (if) future work-meraki. The above mindmap of idea is not something I’m pursuing anymore but bits from it has been taken and it developed in a prototype of camera switch of in and out of a heart model in a space void(without narrative animation, which I would like to include in future proj).







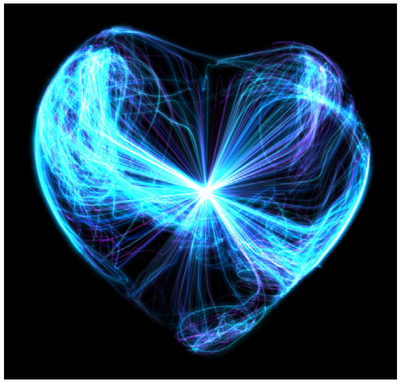
```
import c4d
#Welcome to the world of Python


def main():
    if frame == 1:
        spline = splineobject (2, c4d.SPLINETYPE_LINEAR)
        points = []
        points.append (Pos1)
        points.append (Pos2)

        points.SetAllPoints (points)

        spline.InsertUnder (kaboom)
```

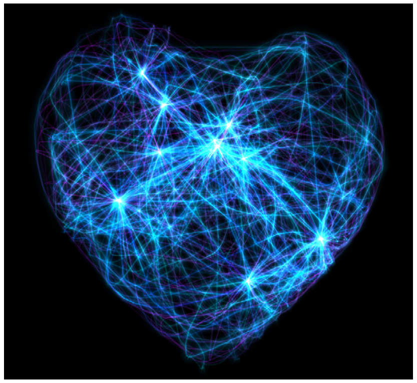
Line 14, Pos. 36

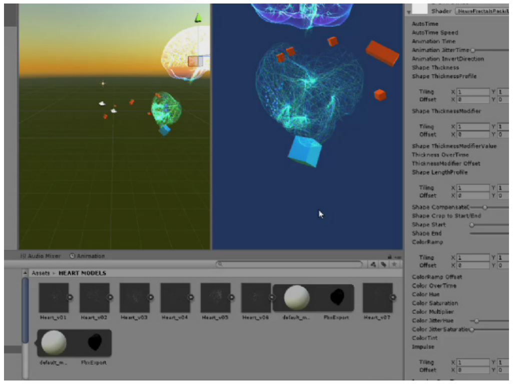
Click on compile to test the script

Spent a good while on making this MAGI logo (for expo) . This was generated through scripting, and it became very heavy, containing tons of splines and objects, which meant lots of crashes, computer hang-over's, restarts , and infinite render times.

This made inspiration for heart model which I use in the project.



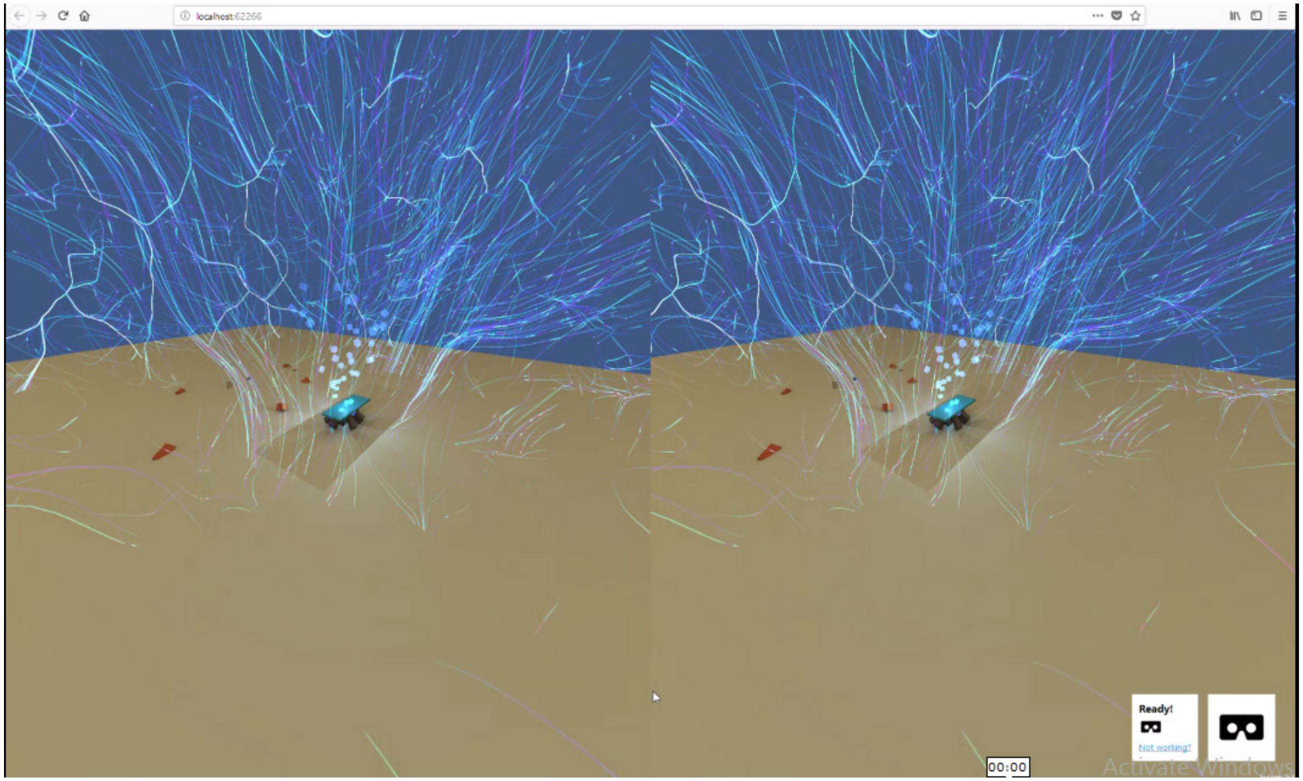




I also use asset packs starcluster and neurofractal by voldemar to help drive the visuals.

Development / \|| tests

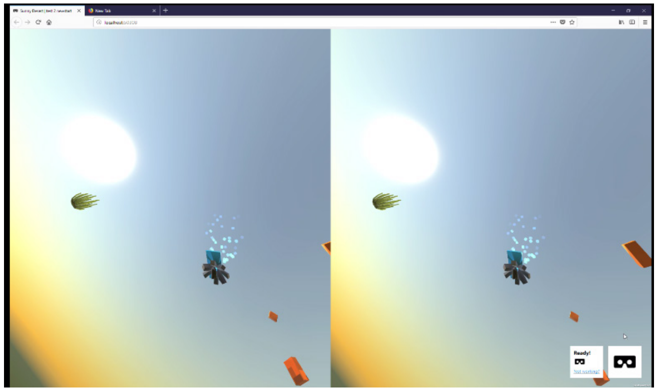
Visual test compaitibility with webVR. (Inside Heart View)

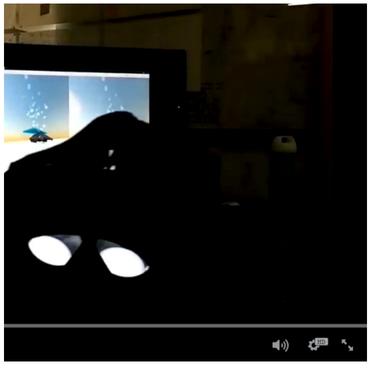


After succesfull compatibilty test of the scene, I further need to get mold this neuron visual inside around the heart mesh and animate it to beat and color and more.

The camera switch from thirdperson(fly targetted) to normal gave a fly to sudden grounding and perception/feeling. In my view , only because we are so used to viewing this reality in a certain way to our head movement. Changing this is really interesting , maybe with some context on VR's ability to play with our perception of reality if experimented and done so.~Digging deeper in this in research methods subject.

Developing the camera Dynamics // (switch between targetted(outside) camera set and normal 360(inside)





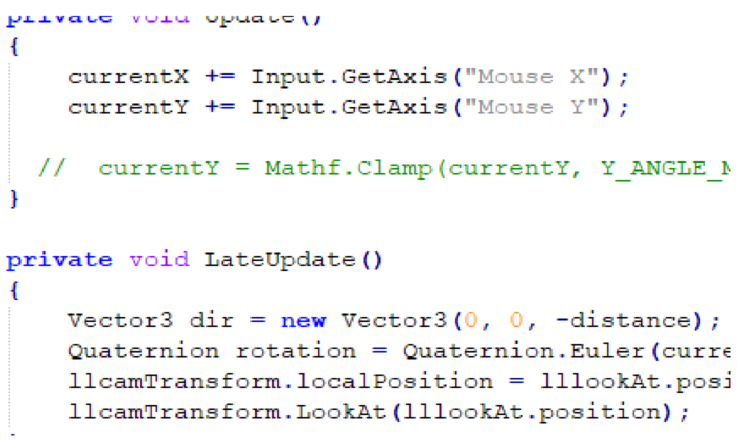
Right now called from keyboard or mouse, need to link it through controller button as well.

Click image to go to video.

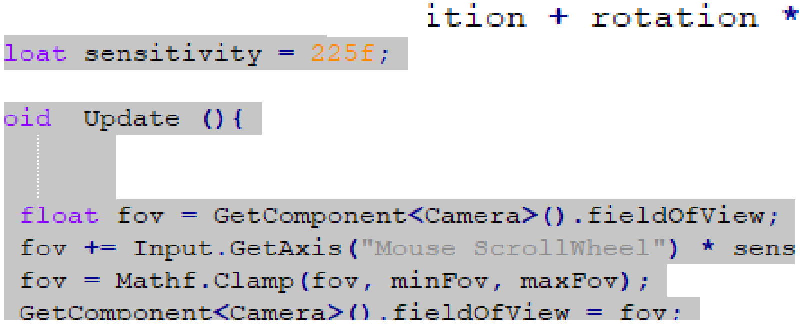
The Glitch?bug : I've got the concept script working-molding around some applied logics. Though for the first switch it works perfectly and on second switch the camera comes out of vr mode, which shouldn't happen. Looking at it I feel the logic behind scripts needs to be re assessed with possibilty of rewriting the whole scripts. It's a bit triicky as it deals with WebVR and taking gyro info fom vr and updating individual cameras for VR and a lot more.

more testing ,making and learning different scripts in project support:

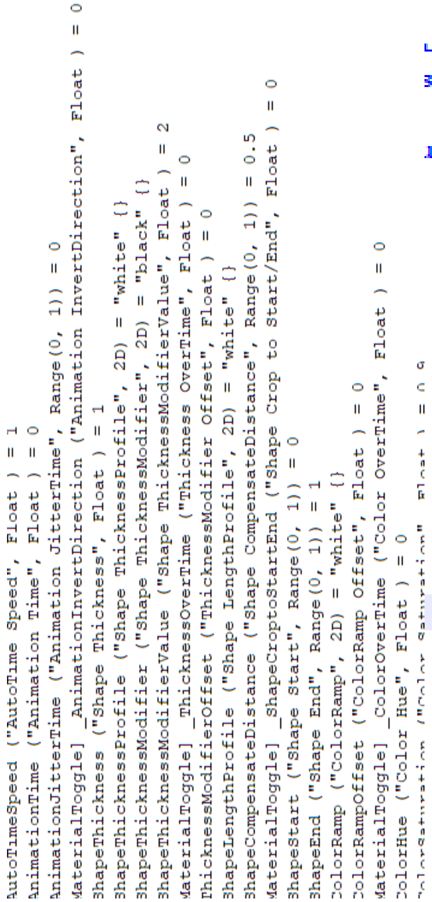
- put camera heirarchy?
- camera switches
- spawner
- camera rotates
- getting input from VR controllers through webvr api
- lots of more bits n snips



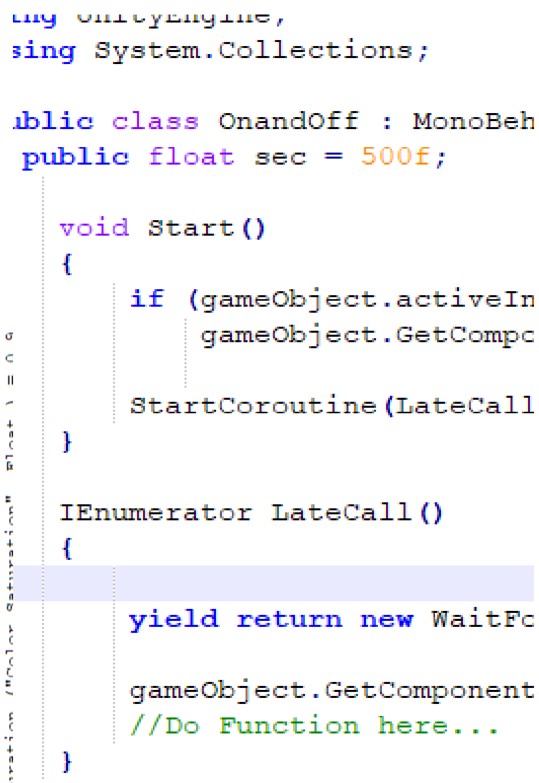
```
private void Update()
{
    currentX += Input.GetAxis("Mouse X");
    currentY += Input.GetAxis("Mouse Y");

  //  currentY = Mathf.Clamp(currentY, Y_ANGLE_M
}

private void LateUpdate()
{
    Vector3 dir = new Vector3(0, 0, -distance);
    Quaternion rotation = Quaternion.Euler(curre
    llcamTransform.localPosition = lllookAt.posi
    llcamTransform.LookAt(lllookAt.position);
```



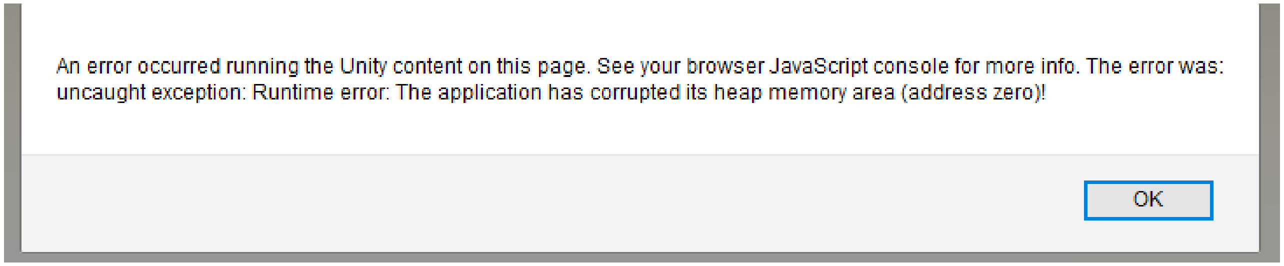
```
ition + rotation *
loat sensitivity = 225f;

oid  Update (){

 float fov = GetComponent<Camera>().fieldOfView;
 fov += Input.GetAxis("Mouse ScrollWheel") * sens
 fov = Mathf.Clamp(fov, minFov, maxFov);
 GetComponent<Camera>().fieldOfView = fov;
```



```
AutoTimeSpeed ("AutoTime Speed", Float ) = 1
AnimationTime ("Animation Time", Float ) = 0
AnimationJitterTime ("AnimationTime JitterTime", Range(0, 1)) = 0
MaterialToggle] _AnimationInvertDirection ("Animation InvertDirection", Float ) = 0
ShapeThickness ("Shape Thickness", Float ) = 1
ShapeThicknessProfile ("Shape ThicknessProfile", 2D) = "white" {}
ShapeThicknessModifier ("Shape ThicknessModifier", 2D) = "black" {}
ShapeThicknessModifierValue ("Shape ThicknessModifierValue", Float ) = 2
MaterialToggle] _ThicknessOverTime ("Thickness OverTime", Float ) = 0
ThicknessModifierOffset ("ThicknessModifier Offset", Float ) = 0
ShapeLengthProfile ("Shape LengthProfile", 2D) = "white" {}
ShapeCompensateDistance ("Shape CompensateDistance", Range(0, 1)) = 0.5
MaterialToggle] _ShapeCroptoStartEnd ("Shape Crop to Start/End", Float ) = 0
ShapeStart ("Shape Start", Range(0, 1)) = 0
ShapeEnd ("Shape End", Range(0, 1)) = 1
ColorRamp ("ColorRamp", 2D) = "white" {}
ColorRampOffset ("ColorRamp Offset", Float ) = 0
MaterialToggle] _ColorOverTime ("Color OverTime", Float ) = 0
ColorHue ("Color Hue", Float ) = 0
ColorSaturation ("Color Saturation", Float ) = 0.9
```



```
ing UnityEngine;
sing System.Collections;

ublic class OnandOff : MonoBeh
 public float sec = 500f;

   void Start()
   {
       if (gameObject.activeIn
           gameObject.GetCompo

       StartCoroutine(LateCall
   }

   IEnumerator LateCall()
   {

       yield return new WaitFo

       gameObject.GetComponent
       //Do Function here...
   }
```



An error occurred running the Unity content on this page. See your browser JavaScript console for more info. The error was:
uncaught exception: Runtime error: The application has corrupted its heap memory area (address zero)!

How much can it handle and memory allotements!!

A major dragger was everytime to test the webvr api with CAMERA dynamics/coding - project needs to be build rather than just running through player as in Unity player it overrides the script for camera headtrack(in vr) and uses in built support mode but webvr derives from script and hence different camera dynamics are possible, every build take 5-20 minutes(minimum only) and till this moment, roughly done around 80-90 builds just to test scripting and settings for development prototype :)

Hosting test:
It was important at this stage to test some builds through a website host using actual internet. I found itch.io to host and test out as github was putting a limit of 25mb for build.

Link to some development builds : https://itch.io/profile/saszer

estio128modnocluster testio101 testio800 testio12

The numbers at end states the ram/memory it heaps in browser to run the game, too much won't let it load in all browsers, too little and your game won't have enough memory to run from.

I suggest, you view testio128 or earlier (which has full star systems , which i've thrown there, to test out how much heaviness can different headsets and overall can handle). try dragging mouse around and middle mouse button and letter 'a' on keyboard to change between camera dynamics, and click vr button which will show if a vr headset is detected. It also has some prototyping sound(which is 3D-affected by space movement- but in very early stages). edit-latest one !

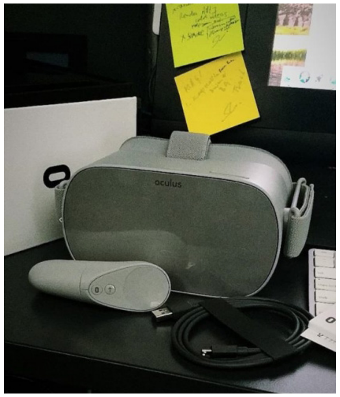
Oculus Go test:

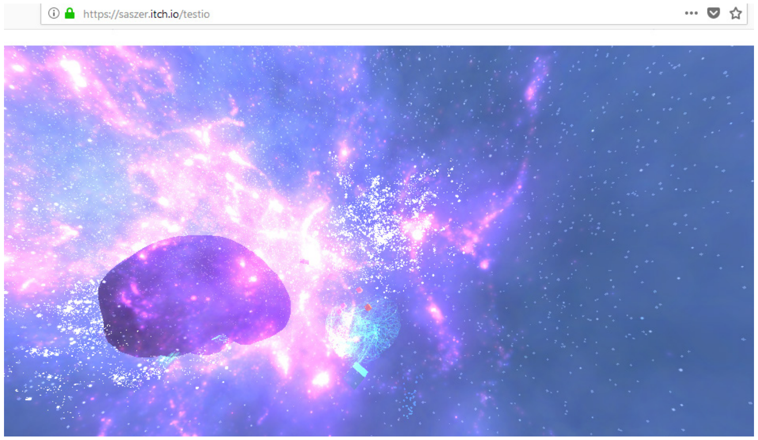
Conculsions after few host build, oculusgo which is a standalone (- andoid) was able to atleast load the game (browser) with around memory allocation of 80-170mb.

HTC Vive tests were straight enough but OculusGo won't run webgl build unless its hosted from HTTP server and hence.

Very very high amount of framedrop and performance issues were revealed from these test which told how much a go browser in 360 can handle.

I also tried ***APK(as an app)*** build inside oculusGo but custom camera dynamic scripts didnt work as it derives through webvr api and will need to be written from scratch for the same.





Dropped framerates - - - Optimizing the heaviness , memory and build

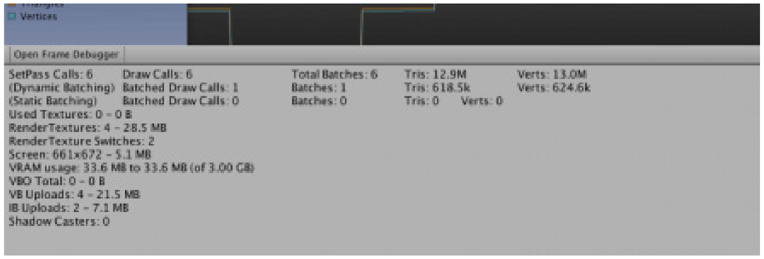
Anything below 90fps is something that a vr headset can't put properly and is not good as a user experience. The current builds in tests were going down to 30fps - even on a pretty good performance system. , in VR it kept showing blackframes in between as it struggled to render the experience.

This meant optimising meshes , scripts , graphics , and more!!-

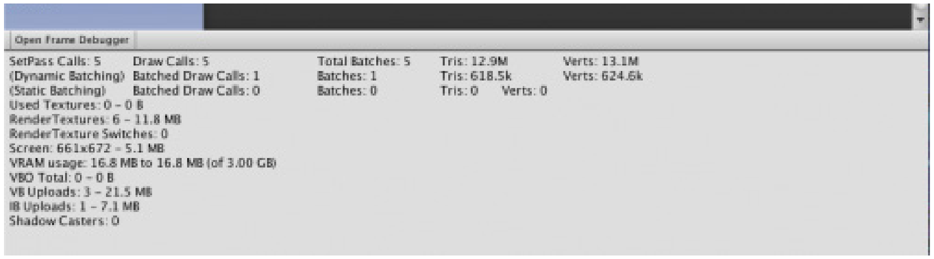
Lightweight Render Pipeline: this is something in built in Unity which is a scriptable render pipeline reducing visual quality a little bit but with high impact on performance. It's a scriptable render pipeline and gives more control over how the game is rendered.

Tried to get this workin but it didn't support VR and custom shaders I'm currently using in project and gives just black screen.
So I reduced the quality through other methods and in doing so I MESSED up the project - camera script stopped working(which took ages of build tests), and visual quality went damn low. Fixing and getting it up all back took some toll!

LWRP test one: After looking and ***modifying script of the shader(neurofractal) I*** was using, I was able to make it with work with lwrp in VR!.



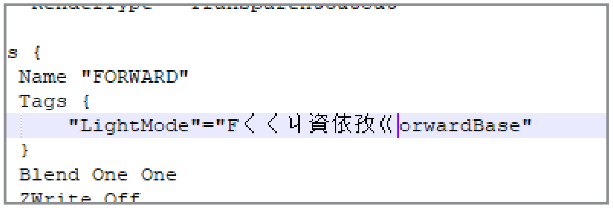
```
SetPass Calls: 6      Draw Calls: 6                 Total Batches: 6    Tris: 12.9M        Verts: 13.0M
(Dynamic Batching)    Batched Draw Calls: 1         Batches: 1          Tris: 618.5k       Verts: 624.6k
(Static Batching)     Batched Draw Calls: 0         Batches: 0          Tris: 0    Verts: 0
Used Textures: 0 - 0 B
RenderTextures: 4 - 28.5 MB
RenderTexture Switches: 2
Screen: 661x672 - 5.1 MB
VRAM usage: 33.6 MB to 33.6 MB (of 3.00 GB)
VBO Total: 0 - 0 B
VB Uploads: 4 - 21.5 MB
IB Uploads: 2 - 7.1 MB
Shadow Casters: 0
```



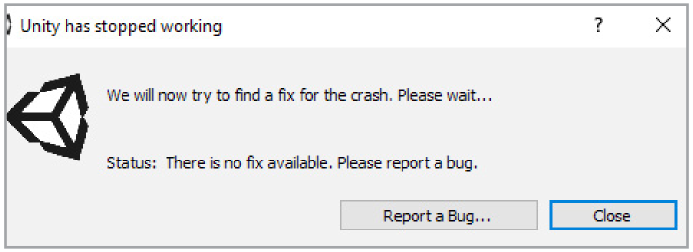
```
SetPass Calls: 5      Draw Calls: 5                 Total Batches: 5    Tris: 12.9M        Verts: 13.1M
(Dynamic Batching)    Batched Draw Calls: 1         Batches: 1          Tris: 618.5k       Verts: 624.6k
(Static Batching)     Batched Draw Calls: 0         Batches: 0          Tris: 0    Verts: 0
Used Textures: 0 - 0 B
RenderTextures: 6 - 11.8 MB
RenderTexture Switches: 0
Screen: 661x672 - 5.1 MB
VRAM usage: 16.8 MB to 16.8 MB (of 3.00 GB)
VBO Total: 0 - 0 B
VB Uploads: 3 - 21.5 MB
IB Uploads: 1 - 7.1 MB
Shadow Casters: 0
```

before lwrp - after lwrp , vram and texture usage decrease by 50%.
Though succesful at first in smaller prototype, I started to notice LWRP started to increase memory usage in bigger project instead of decreasing it, so I kept this on side and jumped on finalising the environment and model//visual. After/along which I can look at optimisation in depth.

a fun one: I opened the pc next day and keyboard was typing in chinese, this meant learning chinese and to code in chinese! , ㅇㅅㅇ
'can we code in chinese?'
spent good time figuring out how to revert back, and then it was right there.



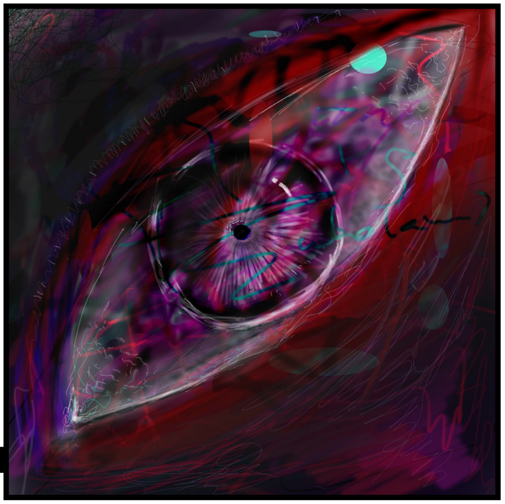
```
s {
 Name "FORWARD"
 Tags {
     "LightMode"="F く く ㅐ資依孜《orwardBase"
 }
 Blend One One
```



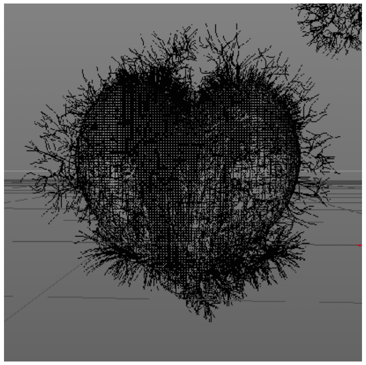
Unity has stopped working

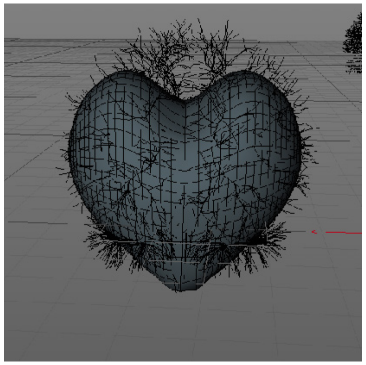
We will now try to find a fix for the crash. Please wait...

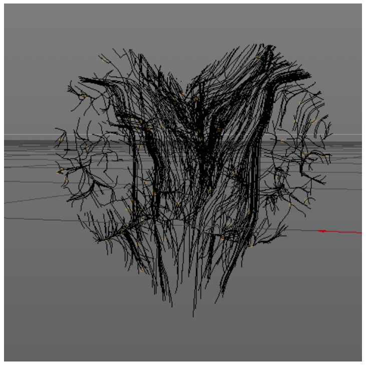
Status: There is no fix available. Please report a bug.

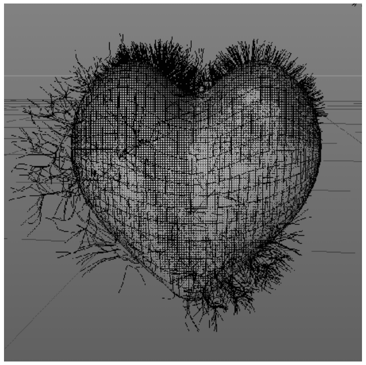


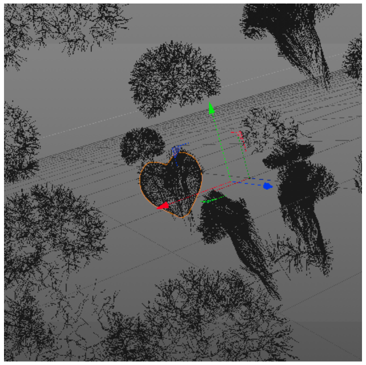
Brain fog 2.0 (photoshop paint)

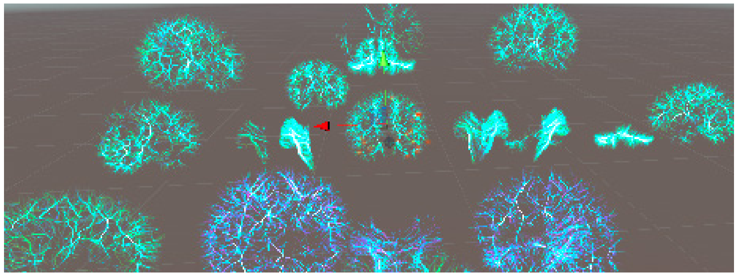












`Modelling the heart , back on the fun side , modelling the dimensions inside heart. A lot of mesh models were made and compared.

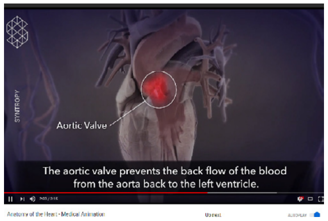
Nervous systems and workings of heart were also looked upon and derived from.

Trees are a major influence and fascination that went into modelling this-chaotic symmetry.

From the point view of the whole(project), how our world -the city- exists within one system - represented by heart?

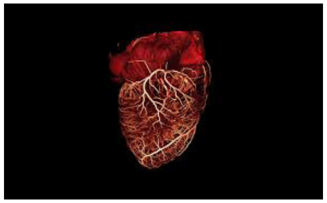
Conflict ~
A constant push and pull between artistic flair and performance . The artist is deriving from a very heavy mesh and performance becomes an issue with it.

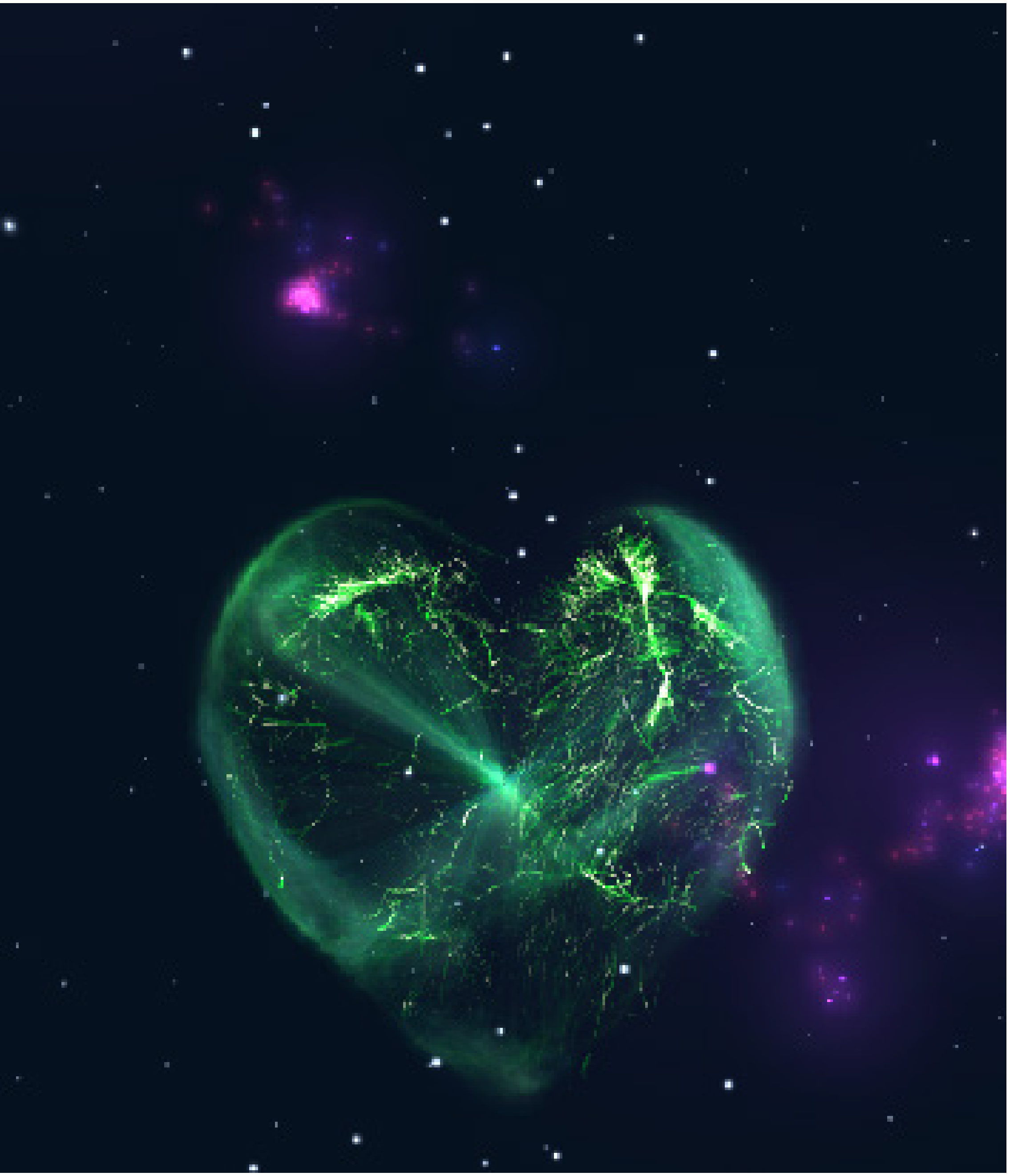
~ in order to find the balance.

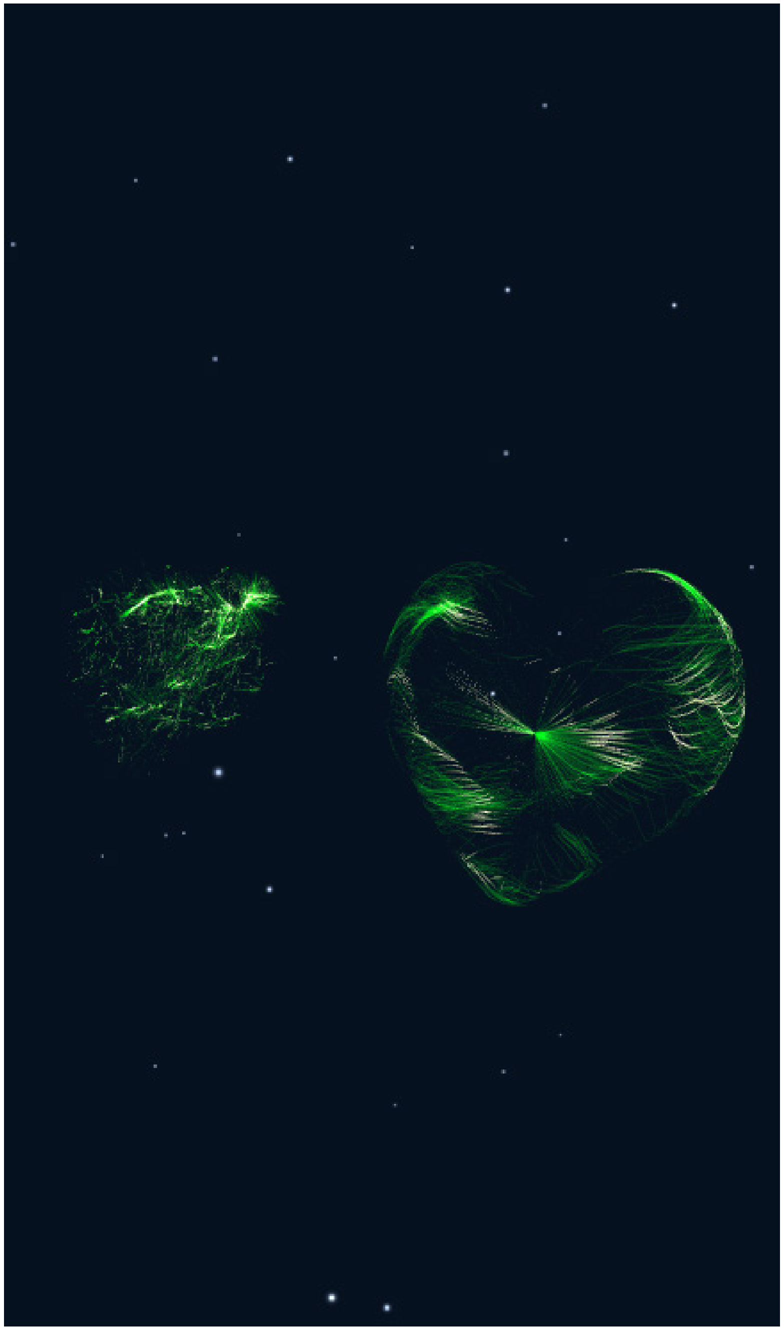


The aortic valve prevents the back flow of the blood from the aorta back to the left ventricle.

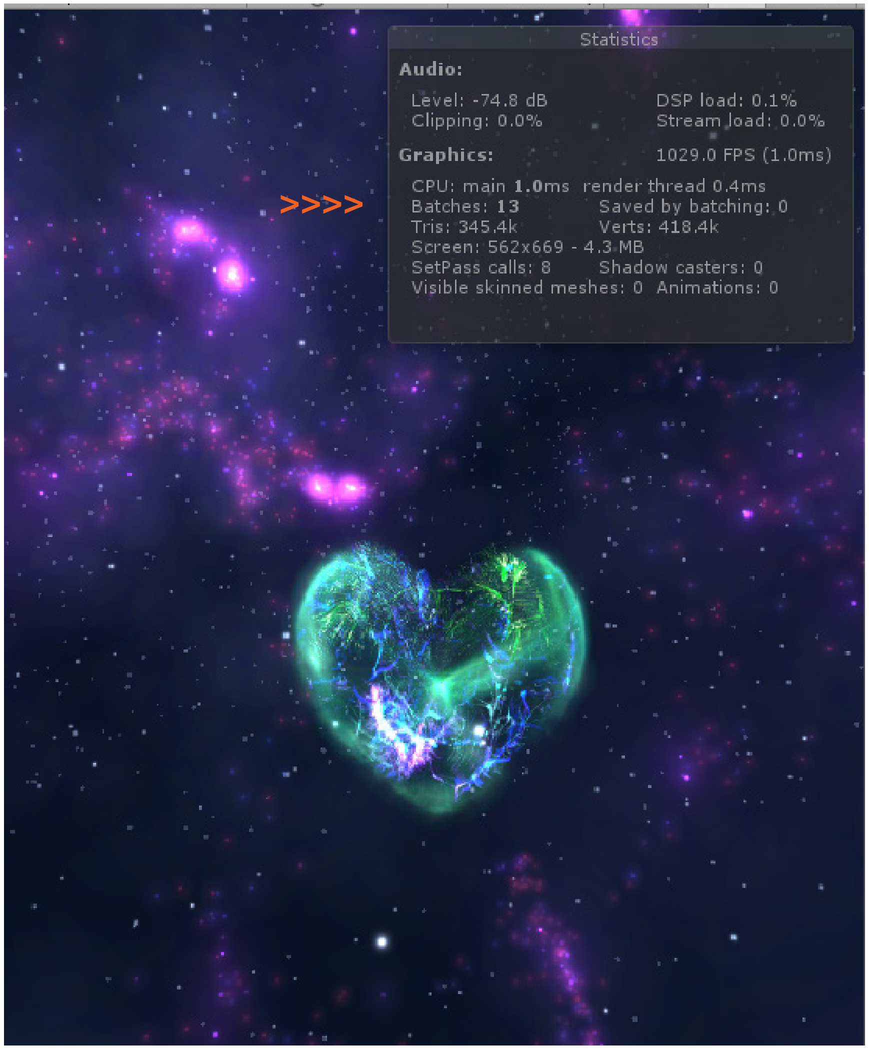
Anatomy of the Heart - Medical Animation





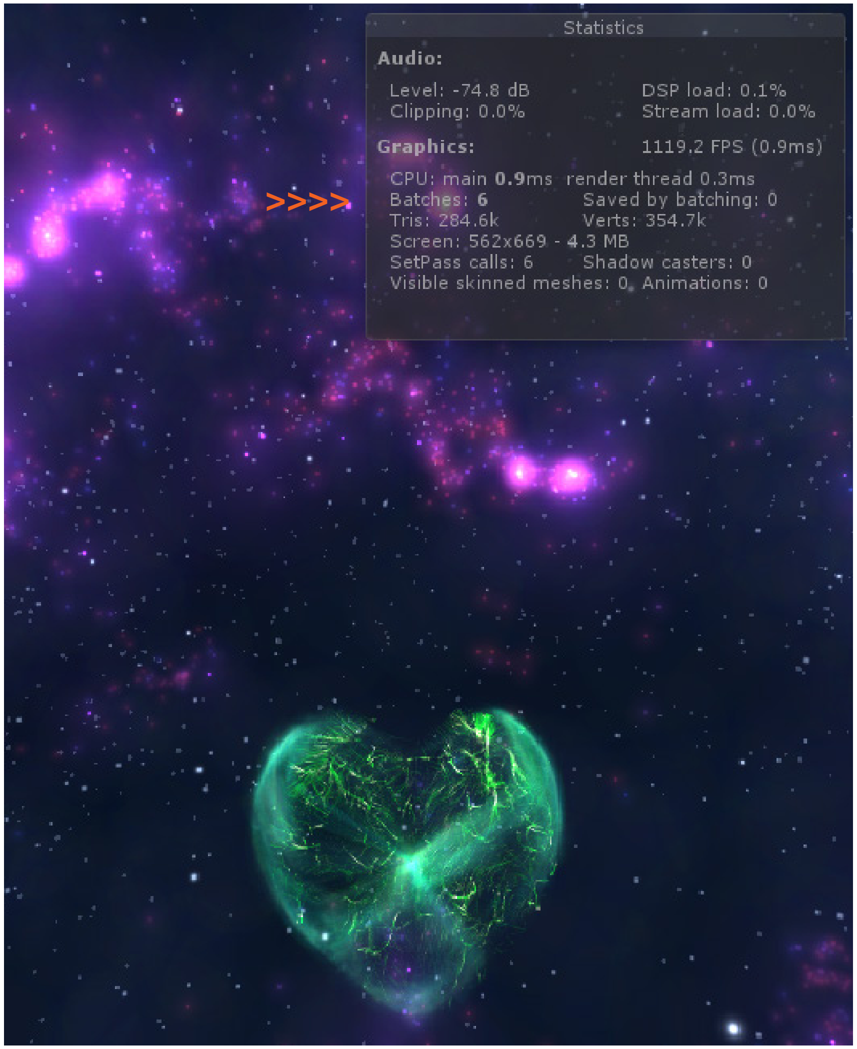


optimising heart mesh / combine batches (image)



```
Statistics
Audio:
  Level: -74.8 dB              DSP load: 0.1%
  Clipping: 0.0%               Stream load: 0.0%
Graphics:                      1029.0 FPS (1.0ms)
  CPU: main 1.0ms  render thread 0.4ms
  Batches: 13          Saved by batching: 0
  Tris: 345.4k         Verts: 418.4k
  Screen: 562x669 - 4.3 MB
  SetPass calls: 8     Shadow casters: 0
  Visible skinned meshes: 0  Animations: 0
```

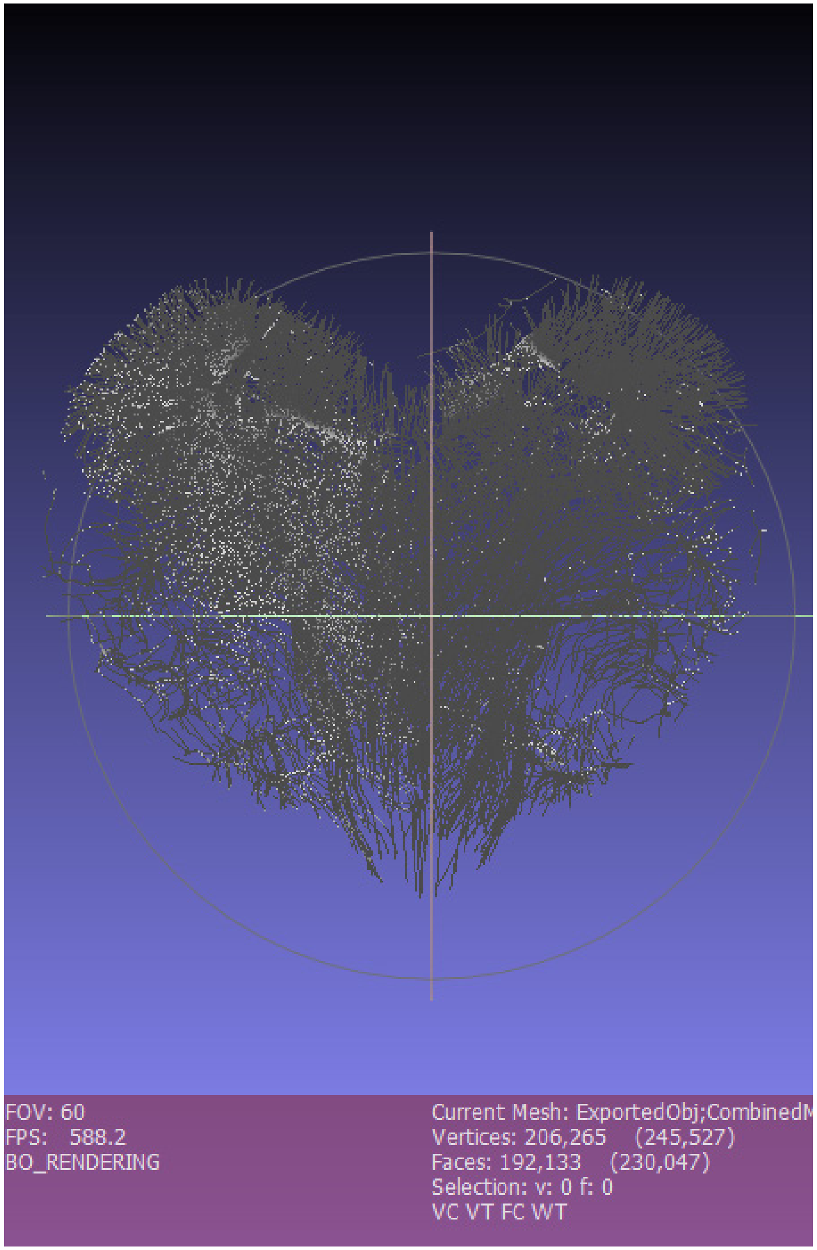
Batches BEFORE combining meshes(heart) : 13



```
Statistics
Audio:
  Level: -74.8 dB              DSP load: 0.1%
  Clipping: 0.0%               Stream load: 0.0%
Graphics:                      1119.2 FPS (0.9ms)
  CPU: main 0.9ms  render thread 0.3ms
  Batches: 6           Saved by batching: 0
  Tris: 284.6k         Verts: 354.7k
  Screen: 562x669 - 4.3 MB
  SetPass calls: 6     Shadow casters: 0
  Visible skinned meshes: 0  Animations: 0
```

Batches AFTER combining meshes(heart) : 6

combining the meshes to reduce the number of batches(process run to render each frame). // increased framerate // less number of processes/draws for each frame. Deep profiling and frame debugger was also studied and looked at to see how each frame is rendered and what processes happen,



```
FOV: 60
FPS:   588.2
BO_RENDERING
```

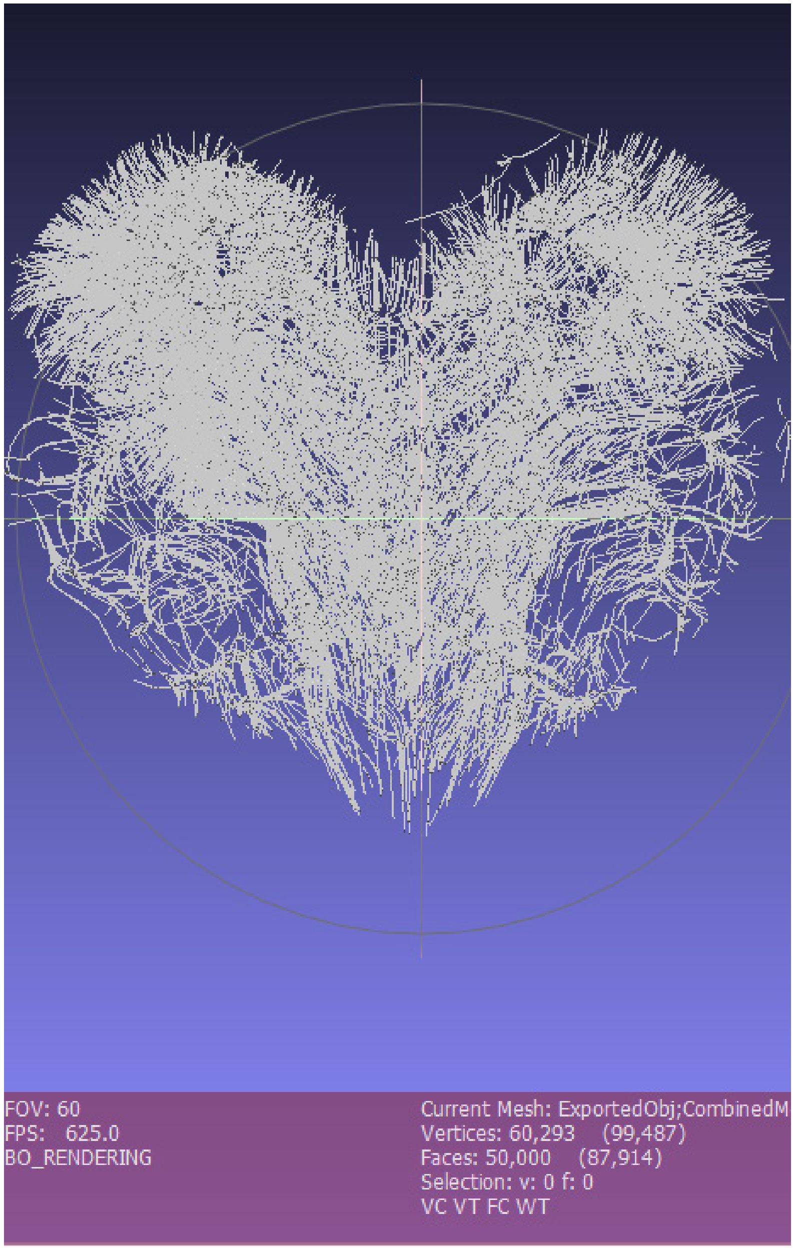
```
Current Mesh: ExportedObj;CombinedM
Vertices: 206,265    (245,527)
Faces: 192,133    (230,047)
Selection: v: 0 f: 0
VC VT FC WT
```

even further:

reducing the model complexity.

MESHLAB is a great tool to optimise meshes to reduce number of faces,vertices etc.

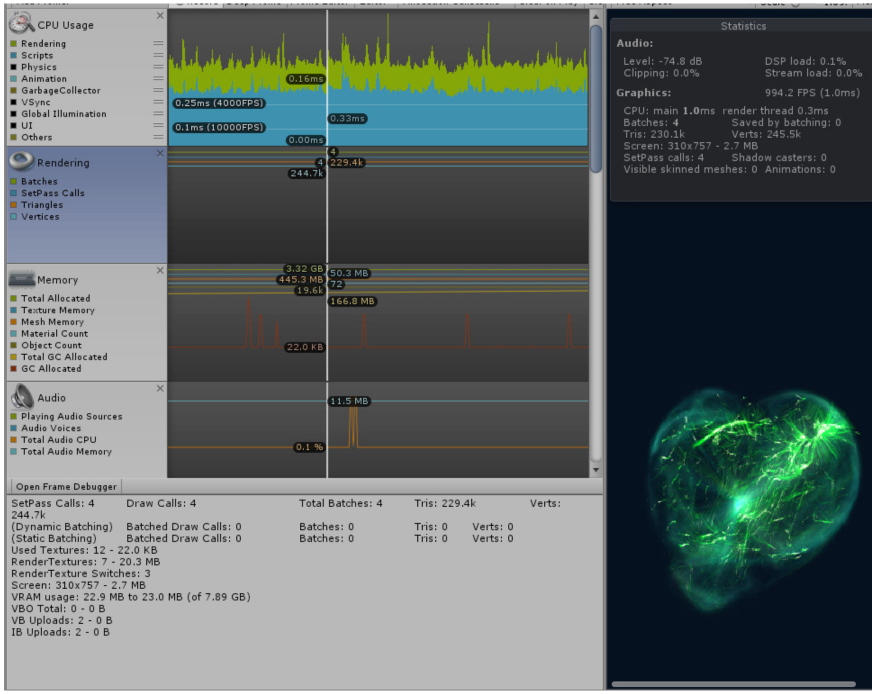
*but I didnt use the lower poly optimised mesh because it lowered visual quality and as I was at a stage where i was able to manage performance through other methods than lowering the aesthetics.



```
FOV: 60
FPS:   625.0
BO_RENDERING
```

```
Current Mesh: ExportedObj;CombinedM
Vertices: 60,293    (99,487)
Faces: 50,000    (87,914)
Selection: v: 0 f: 0
VC VT FC WT
```

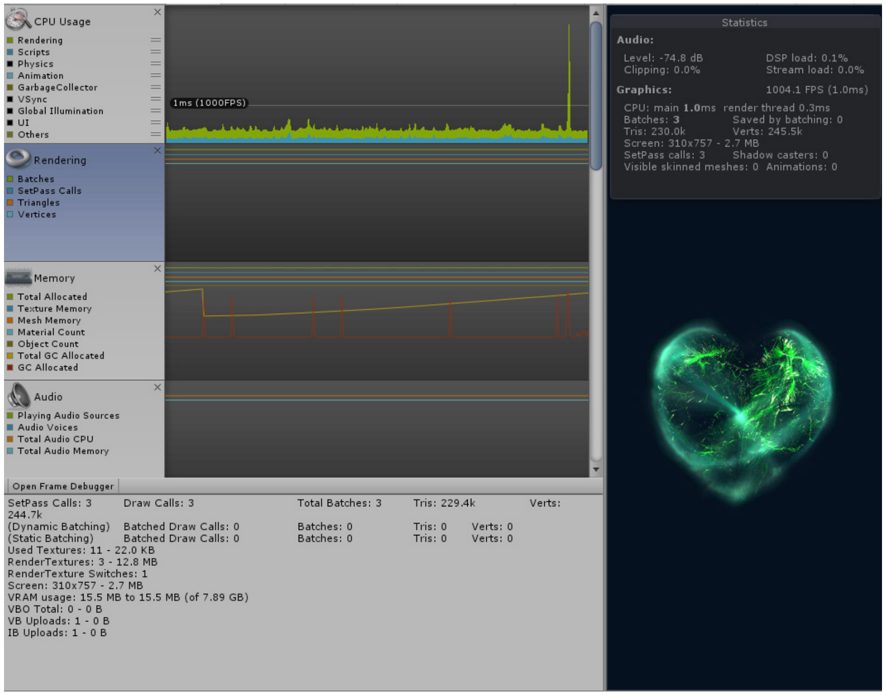
Lightweight Render Pipeline haywire!!
(image)



```
Statistics
Audio:
  Level: -74.8 dB            DSP load: 0.1%
  Clipping: 0.0%             Stream load: 0.0%
Graphics:                    994.2 FPS (1.0ms)
  CPU: main 1.0ms  render thread 0.3ms
  Batches: 4            Saved by batching: 0
  Tris: 230.1k          Verts: 245.5k
  Screen: 310x757 - 2.7 MB
  SetPass calls: 4      Shadow casters: 0
  Visible skinned meshes: 0  Animations: 0
```

```
SetPass Calls: 4      Draw Calls: 4               Total Batches: 4    Tris: 229.4k      Verts:
244.7k
(Dynamic Batching)    Batched Draw Calls: 0       Batches: 0          Tris: 0    Verts: 0
(Static Batching)     Batched Draw Calls: 0       Batches: 0          Tris: 0    Verts: 0
Used Textures: 12 - 22.0 KB
RenderTextures: 7 - 20.3 MB
RenderTexture Switches: 3
Screen: 310x757 - 2.7 MB
VRAM usage: 22.9 MB to 23.0 MB (of 7.89 GB)
VBO Total: 0 - 0 B
VB Uploads: 2 - 0 B
IB Uploads: 2 - 0 B
```

with lwrp :
number of batches : 4
Vram : 23mb
Render Texture : 7 - 20.3mb
AB/IB upload : 2



```
Statistics
Audio:
  Level: -74.8 dB            DSP load: 0.1%
  Clipping: 0.0%             Stream load: 0.0%
Graphics:                    1004.1 FPS (1.0ms)
  CPU: main 1.0ms  render thread 0.3ms
  Batches: 3            Saved by batching: 0
  Tris: 230.0k          Verts: 245.5k
  Screen: 310x757 - 2.7 MB
  SetPass calls: 3      Shadow casters: 0
  Visible skinned meshes: 0  Animations: 0
```

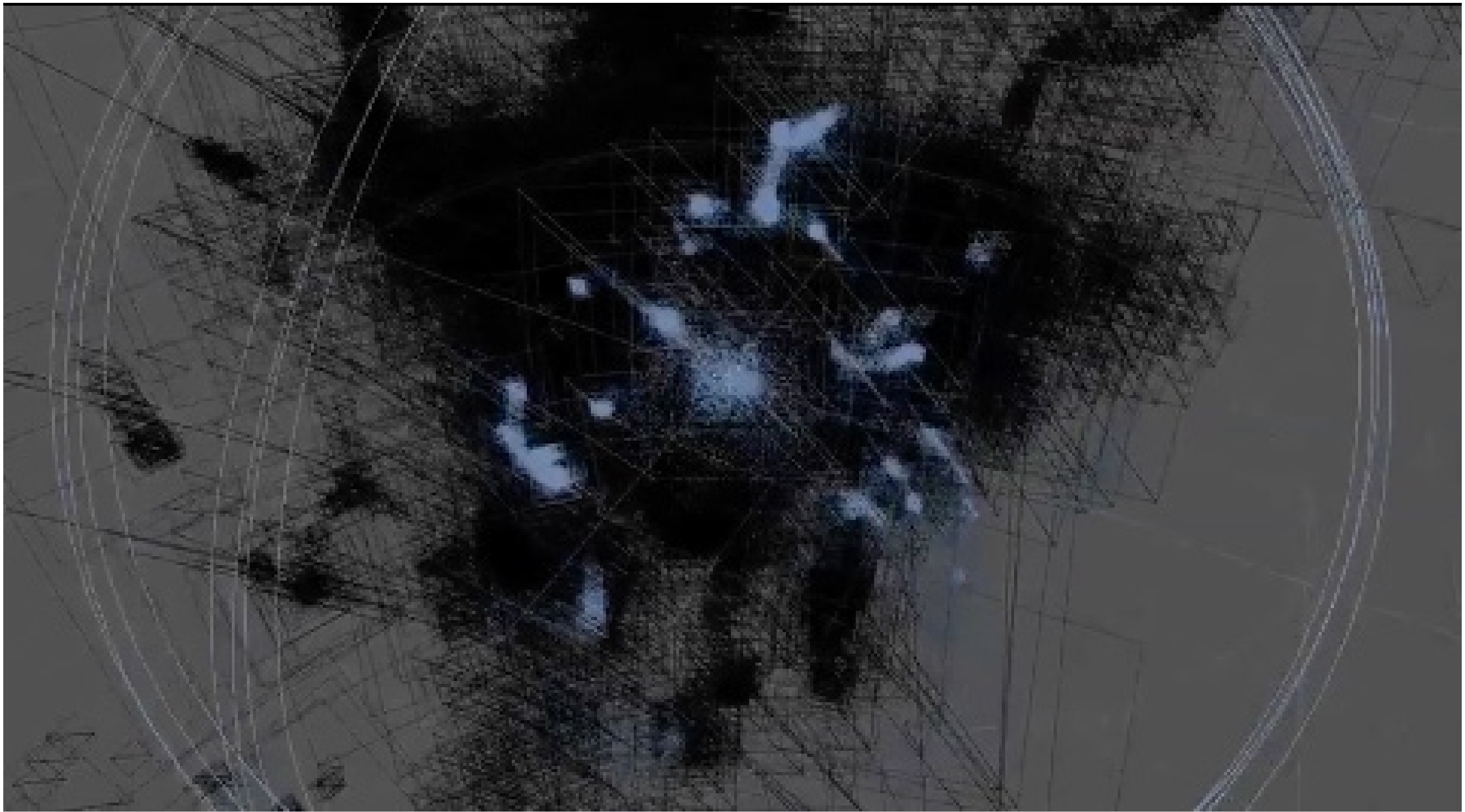
```
SetPass Calls: 3      Draw Calls: 3               Total Batches: 3    Tris: 229.4k      Verts:
244.7k
(Dynamic Batching)    Batched Draw Calls: 0       Batches: 0          Tris: 0    Verts: 0
(Static Batching)     Batched Draw Calls: 0       Batches: 0          Tris: 0    Verts: 0
Used Textures: 11 - 22.0 KB
RenderTextures: 3 - 12.8 MB
RenderTexture Switches: 1
Screen: 310x757 - 2.7 MB
VRAM usage: 15.5 MB to 15.5 MB (of 7.89 GB)
VBO Total: 0 - 0 B
VB Uploads: 1 - 0 B
IB Uploads: 1 - 0 B
```

without lwrp :
number of batches : 3
Vram : 15mb
Render Texture : 3- 12.8mb
AB/IB upload : 1

other factors like tris , render switches , fps, etc were also higher in LWRP(which was excpected to be opposite of)

After good amount of tests, my conclusions that Lightweight Render Pipeline was increasing the framerates than lowering it strengthened. It increased the number of vertices/faces / batches etc. It could be probably because of custom shaders the project is using but this needs analysis.
So LWRP was taken out. (□o´(□┐□(□,□)□

The Mesh , art we don’t get to see // Creating the star world and optimisation.



click to go to video. // symmetry in chaos

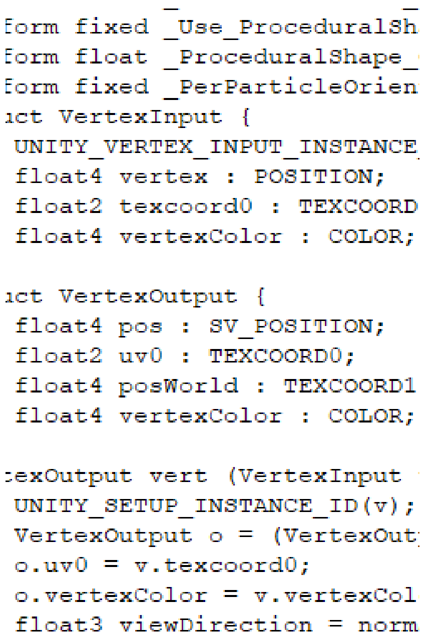
keywords: Profiling (checking what is taking up tme/render) ||| frame debugger - seeing how every frame is rendered. ||| gpu intancing ||| vsync off, anti aliasing , fastest quality settings, null shadow ||| combining meshes with similar properties. ||
*Batches - number of process cpu does to render one frame.

To optimise for the heavy mesh of the starworld, the whole (star)system(prefab) was broken and teared apart.
Instead of a single huge fixed mesh with heavy amount vertices, It was broken to a single star and called many times from one mesh. (the same was done for fog and cluster).This also reduced mesh data it has to load up Though this meant instead of calling out a single (huge)mesh (earlier) , now it has to call many stars individually which increased number of batches. These were reduced by: 'saved by batching' in unity, in which meshes with same properties gets batched together, so it still got rendered in a single batch(/process).

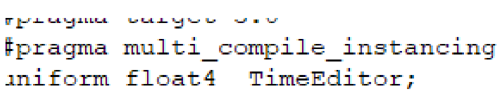
On the runtime a random position script is run for the starsystem to scatter the stars which gives a random new cluster every time its loaded.

CUSTOMISING SHADER SCRIPT : to make it compatible with 'saved by batching' - combining same mesh with same material in a single process instead of rendering seprately in different processes. |||| I'm using starcluster asset which didn't have support for batching , so I studied the shader script and recoded the shader script to support batching. (via)

So now we had a lower poly and customisable starsystem with even reduced number of batches.



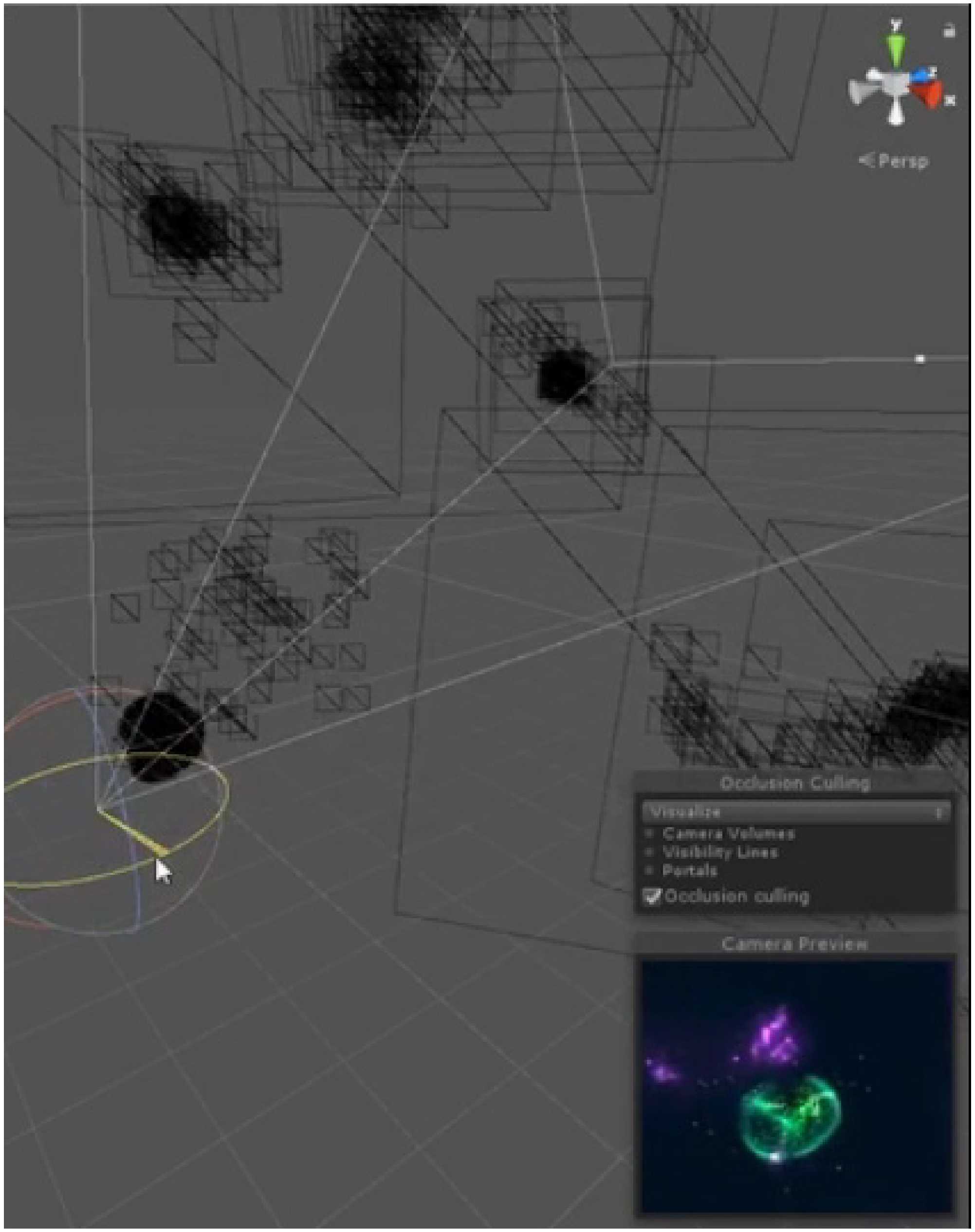
```
form fixed _Use_ProceduralSh
form float _ProceduralShape_
form fixed _PerParticleOrien
uct VertexInput {
 UNITY_VERTEX_INPUT_INSTANCE
 float4 vertex : POSITION;
 float2 texcoord0 : TEXCOORD
 float4 vertexColor : COLOR;

uct VertexOutput {
 float4 pos : SV_POSITION;
 float2 uv0 : TEXCOORD0;
 float4 posWorld : TEXCOORD1
 float4 vertexColor : COLOR;

:exOutput vert (VertexInput
 UNITY_SETUP_INSTANCE_ID(v);
 VertexOutput o = (VertexOut
 o.uv0 = v.texcoord0;
 o.vertexColor = v.vertexCol
 float3 viewDirection = norm
```



```
#pragma target 3.0
#pragma multi_compile_instancing
uniform float4 TimeEditor;
```

GPU Intancing // spawning 100000 objects without meshes!!
I did a small side project(also in development folder/unityprojects) [followed tut.link] - where you derive lots of objects but without any mesh! , it uses a direct render in a single batch(same mesh and material). After spending a while on it, i realised though it decreased my rendertime to almost null it increased my cpu usage to a spike which resulted in even lower framerates than before. so this approachwas chucked out though through, I learnt proper GPU instancing(drawmeshinstanced) and huge potential it holds.

the art of culling,



☑ Occlusion culling

Culling is a feature that disables rendering of objects when they are not currently seen by the camera .

The big clustered single mesh was broken to single system's so they can be identified as individuals and can be culled.

There are also certain set of rules which should be followd to make it work correctly, like for static and dynamic objects.

If an object is moving , and you set it as static, and cull it, it will cull but sometimes because camera thinks it's static, a part of graphic of it might just stick there even if object moved away.

A trick around it can be to use heirarchy and put it under a parent object, which can be set to 'not static' and the child object can be set to 'static ' to meet whatever requirements

Also to note, the star meshes are created from a single instance at runtime , still culls!

cullin ,
<click to go to video.

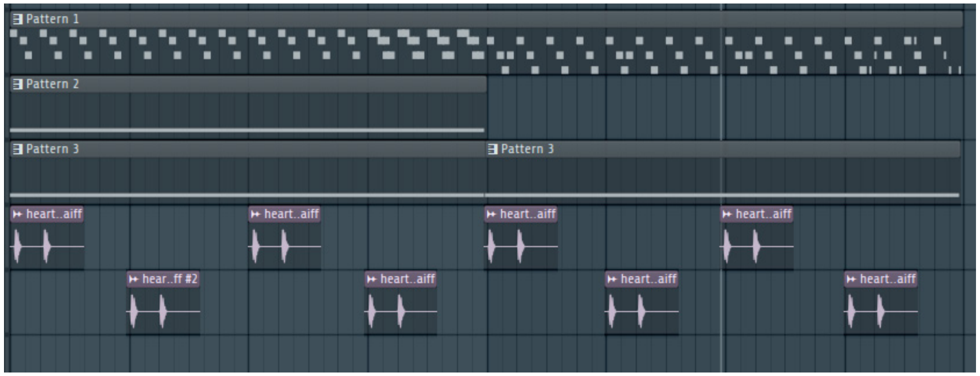
A side visual(done in c4d)[in sides

Aural

The idea of project started as being an immersive music experience, and as said-sound may break or make an experience, here it plays a role of a lot more.

We have discovered and derived 3D graphics from 2D, in sound we went mono to stereo. Now we are exploring Immersive 3D visual, and thus comes along a need and a side to explore, an immersive 3D sound.

It’s important that together it still plays as a whole to be a music , (for this doppler effect was reduced, because it messes with pitch) but still be individual in their spaces.
Unity became a mixer and part of a DAW(Digital Audio Workstation)



Audio programmed in Fruity Loops

A quick demo audio.

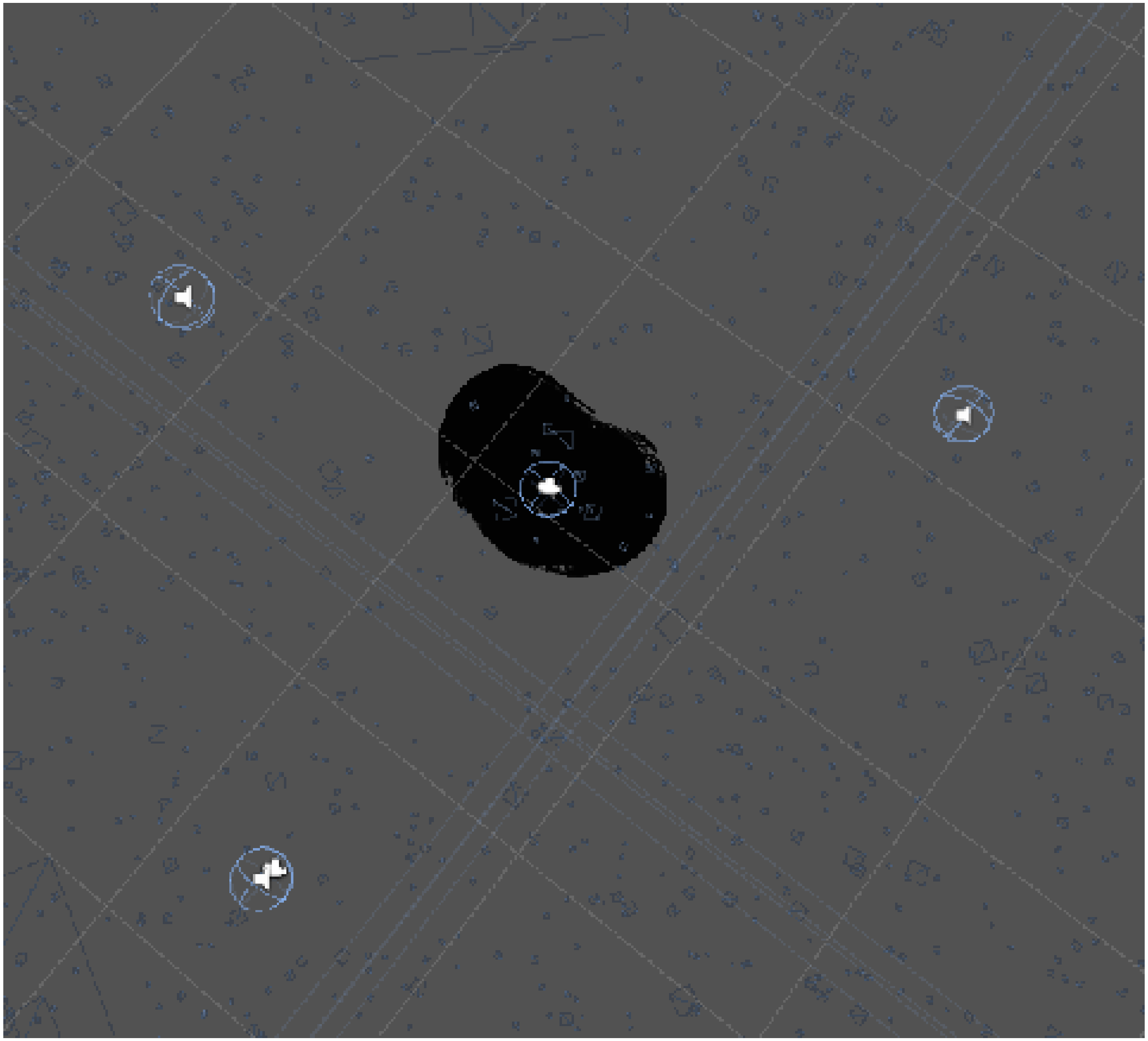
An quick demo track was generated to test out the implementations.

It consists of 4 audio layers,

a synth,
a pad
a bassy gated pad drone
a heartbeat

(view audio files in document folder)

Listening to individual files/stems helps in making out of what’s happening in next sections.



Initial tests ,

The four audio stems were placed as shown in the picture, with heartbeat in the middle.

They are space affected, that is influenced by where the camera is.

Not noticed at firsts tries maybe, because it plays as a whole, but if listened to the mix it somewhat depends on the user.

A user may decide to take 360 turn intensively // lost in space visual and to music, in such they'll experience the song mix panning around their head in 360, also depending on the speed of their movement.
This adds a subtle dimension of audio interaction.

Though more thought needs to be given to achieve a mix of par because now it is influenced by user, limitations and creative placement of audio sources and strategic planning is required.

Like a thumb of rule in a better bass mix is to keep it(low-end) in center, ie- not panned to right or left so that it hits you right at face and feels strong.
The heartbeat which can act as a rhythm/bassdrum is centered here, (also possible because our camera dynamic is inverted).

<<CLICK to go to VIDEO .

Going Haywire // symmetry in chaos.

every star gets its own voice .. ***click image to go to video.***

I did some quick tests where with every star spawned had an audio attached, the result was a blend of 3D ambiance which is affected by head rotation//camera placement in the scene. This can be used in really awesome ways to create atmospheric drones and environemt and a very fluid 3D apce which can fully immerse one deeply (mindfulness-also exploring in research design).

Right now (for initial/in the video above) the audio source was same for all. Further ideas from peers(Janet says> what if they have individual audio sources?) . A really interesting concept that is proposed to further explorations. Though this will come with some performance barriers , it can create an orchestra of .... something.

Future Directions ~

Binaural audio : In 1839, Heinrich Wilhelm Dove discovered that playing a specific tone in one ear and a slightly different tone in the other generates a binaural beat. Specific Binaural beats can be used to put a mind is different states. //(further exploration in research)

Binaural example:

Left Ear	Result	Right Ear
100 hz	13hz	113 hz

taking it further,
- If every star particle(object) is made to have a slightly different hz, It'll be a web of binaural audio which one can move along. Though it'll break binaural effect when a star is somewhere in middle and projecting to both ears.
- hz is basically - pitch of sound. Turning on doppler effect in 3D audio - pitch is then manipulated with distance (imagine how a racing car sounds like when it pasts you to understand).

This is a test ideation which needs more analysis ~ to be explored further.

Conclusions

The project delivers a prototype which to be further polished is one of kind unique 360 vr experience.

A space to be: It offers a unique immersion experience with a unique camera dynamic. The switch between camera dynamic offers and holds a space to be explored more and how the perception of our vision to our used to headmovement can be questioned.

3D audio adds an extra dimension to the whole experience, giving it more immersiveness and mindfulness and a unique way as a user to interact with the aural perception/mix.

This can pave way to unique experiences, delivering mindfulness , a floating sensation which often people confuse as dizziness!? , a narrated story, a unique way to experience, create and consume music, a meditative with a binaural audio experience, an out of world exploration.

A narrative/

Further visions of the project sees it as a narratived story telling/music experience with subtle amounts of interactivity. It also questions and paves way to a new approach to mixing music, which can be integrated really well with a focused audio Visual experience.

A VR devs rescue

It tries to answer the query on multiple platform support for VR via using WebVR, exploring it’s potentials and workarounds. WebVR being in it’s infancy , has a lot more to offer and this project explores a very unique territory within it.

personal devs:

I learnt Unity(totally new) , optimisation, 3D integration , a long lost love to create scripts and logics , VR , inputs, oculus go , htc vive, shaders , WebVR api , and a workflow hich can be used to create some unique creative audio visual experiences/ games/ interactives and more.

visit saszer.itch.io
for the experience over WebVR.

It has development works and tests (might delete as it progresses) . The latest one , currently ‘ ‘ test-withaudio ’ should be the version you wanna look at.

If you have a VR device connected, you’ll see switch to Vr option, if not you can scroll
around on screen as well.

Controls :

-On screen ,

Mouse Right / Keyboard ‘a’   : to switch between cameras
Scroll for time travel ;)
mouse axis for move arounds

-With VR,
working with oculus go tests: controller trigger to change camera. (buggy) .

headphones recoommended.

BEST WAY RIGHT NOW >>

A local HTML version can be run through index.html file in adjacent folder.

*needs proper controller inputs, bug fixes, proper audios, in and outs.

References

1. < https://forum.unity.com/>
2. Light Weight Render Pipelines < https://forum.unity.com/threads/lwrp-4-0-0-preview-is-out.562291/>
3. WebVR assets, < https://github.com/mozilla/unity-webvr-export>
4. VOLDEMAR , starcluster and Neurofractal : https://assetstore.unity.com/search?q=voldemar&k=voldemar