

COMPUTER GRAPHICS & 3D

SUPERNOVA PARTICLE SYSTEM

AIM

- Reproduce the scene of a supernova
- Explosion of debris represented as a particle system
- Use shaders and technologies studied to model the physical behaviour of the particle

PROJECT INTRODUCTION

- A particle system is a convenient representation of a natural phenomena
- The natural phenomena to be reproduced is explosion of debris of a supernova

SUPERNOVA

- A supernova is a astronomical event that occurs during the last stellar evolutionary stages of a star's life, whose destruction is marked by one final explosion. This causes the sudden appearance of a "new" bright star called *pulsar*.
- The explosion sweeping up an expanding shell of gas and dust, which is observed as a supernova remnant.
- Most supernovae are triggered by the sudden gravitational collapse of a massive star's core.

PROJECT MODEL

- The project will be implemented using WebGL
- *explosion* and *debris* particle will be managed through **shaders**
- The scene surrounding *explosion* and *debris* will be represented using **three.js library**

THREE.js

- Three.js is an OpenSource JavaScript library used to create and display animate 3D computer graphics in a browser. Three.js uses WebGL.
- Three.js allow to create complex animations and system that may be much difficult using only javascript
- A well explained documentation is at: <https://threejs.org/docs/>

IMPLEMENTATION

Shaders

- Vertex Shaders
 - Attributes
 - particle starting position
 - particle size
 - particle trajectory
 - time offset (for representing better explosion)
 - Uniforms
 - time t
 - radius
- Fragment Shaders
 - Color
 - Texture

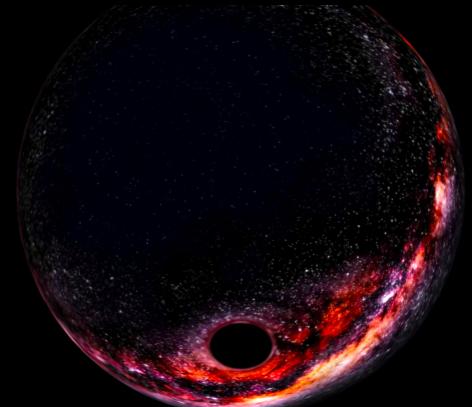
Three.js

- initialize scene
- camera
- light
- space box containing the scene
- geometry
- texture
- star
 - geometry
 - texture
 - glow of star
- geometry
- material
- Audio

SCENE

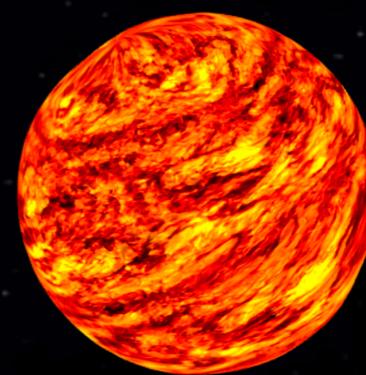
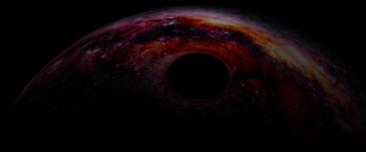
Space Box

- *SpaceBoxGeometry*
 - *spherical and static*
- *MeshBasicMaterial*
 - *space texture style*



Star

- *StarGeometry*
 - *spherical*
- *ShaderMaterial*
 - *displacement on surface (with noise and bump mapping)*
 - *texturing*



SCENE

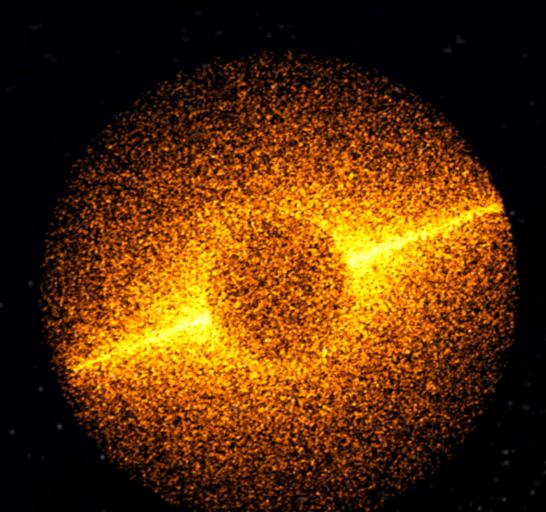
Glow Star

- StarGlowGeometry
 - *spherical (bigger than star)*
- ShaderMaterial
 - *intensity*
 - *color*



Explosion Debris

- BufferGeometry
 - *defined as a set of N vertices with attributes:*
 - *position (theta, phi angles)*
 - *customSize*
 - *timeOffset*
- ShaderMaterial
 - *moves positions to look like an explosion*
 - *texturing*
 - *color*



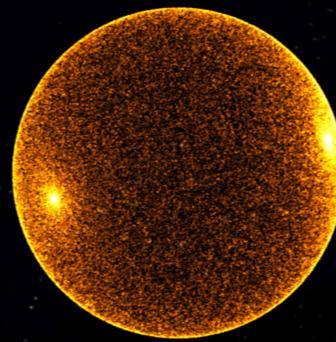
EXPLOSION DEBRIS

- For simplicity, the explosion of debris has the form of sphere that increase radius with time

- $x = (r + \text{time}) * \sin(\theta) * \cos(\phi)$

- $y = (r + \text{time}) * \sin(\theta) * \sin(\phi)$

- $z = (r + \text{time}) * \cos(\theta)$



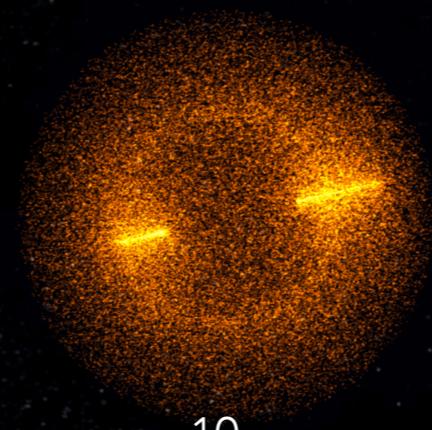
- But doing so does not give the impression of an explosion because the sphere inside remains empty

- One solution is to introduce a *timeOffset* to fill the sphere during the explosion

- $x = (r + \max(0, \text{time} - \text{timeOffset})) * \sin(\theta) * \cos(\phi)$

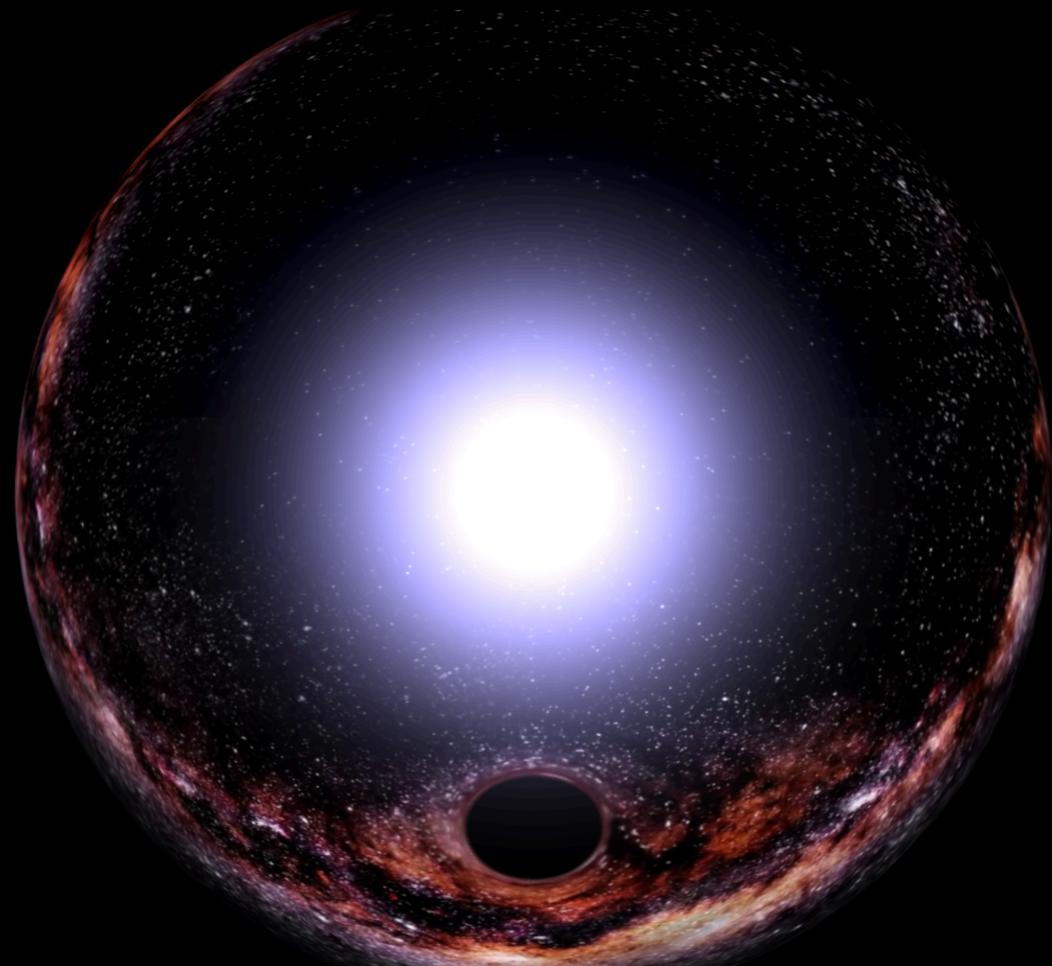
- $y = (r + \max(0, \text{time} - \text{timeOffset})) * \sin(\theta) * \sin(\phi)$

- $z = (r + \max(0, \text{time} - \text{timeOffset})) * \cos(\theta)$



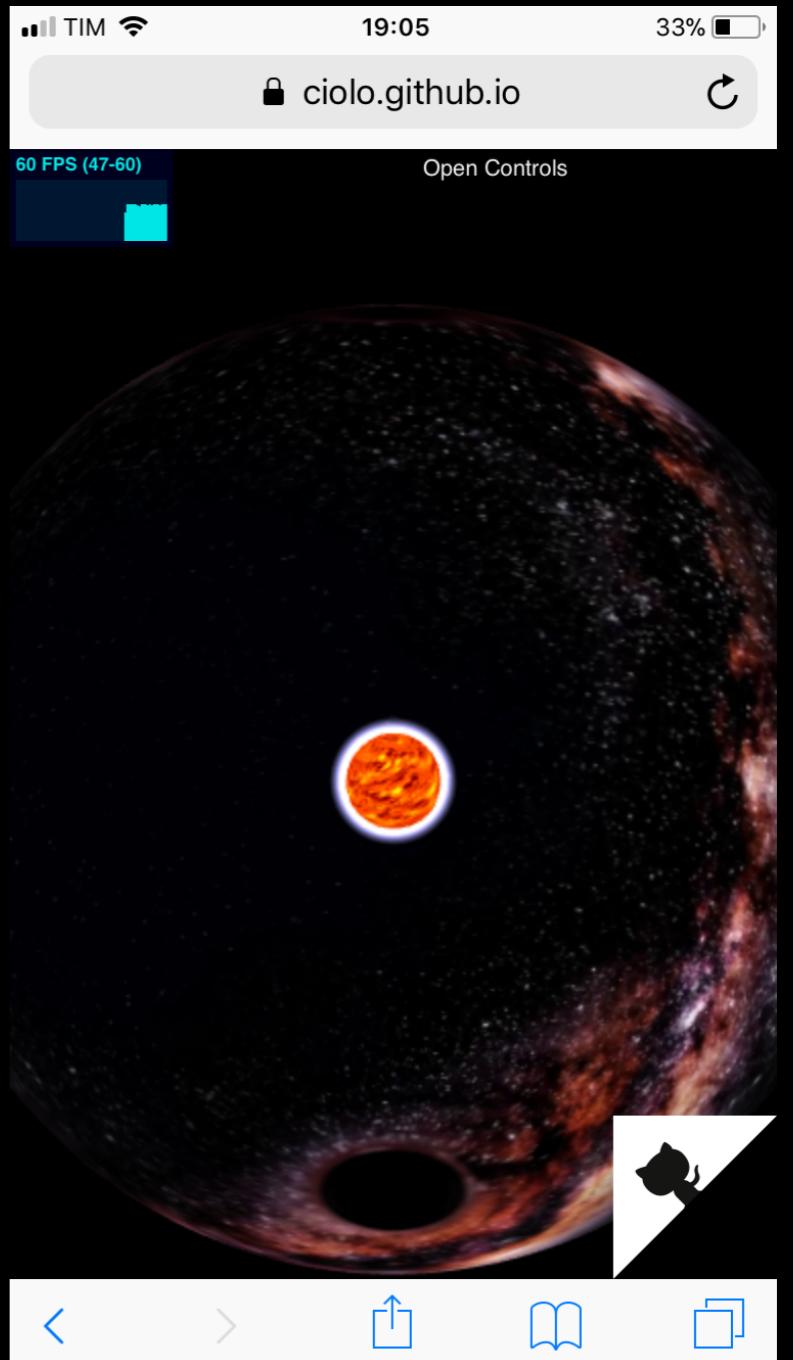
PULSAR

- Like i said before, a supernova bring the "birth" of new star called *pulsar*. This is substantially a star of light, precisely a neutron star that emits a beam of electromagnetic radiation.



MOBILE PORTABILITY

- The project has been made mobile ready with touch event controls and great performances.
- Even an older generation devices ~60 fps are rendered.



PERFORMANCES

- Using a MacBook Pro and Firefox with reference:

Explosion Particle	Fps
50K	60
100K	60
150K	60

- Using an iPhone 6s and Safari with reference:

Explosion Particle	Fps
50K	60
100K	60
150K	60

CONCLUSION

- Link to demo example:

<https://ciolo.github.io/Supernova/>