

# 3D Music Library browser



## The Idea

I wanted to do something to showcase the different aspects of the engine I've been working on this whole semester. One night I was browsing my music library for new music and thought how cool it would be to fly through my music library in 3D, organised into e.g. genres and artists and hear the music in each group as I flew by. That night I didn't really go to sleep...

## How does it work?

The program reads a music library (directories containing music files), builds a database with all the songs and stores the information about every song. It then builds 3D clusters for each of the genres. Every cluster contains blocks representing an artist, the height of the block is relative to the number of songs available for that artist. After the environment has been initialized, random songs are chosen to be played from each block. Now as the user travels around he or she can hear the songs playing in 3D. So if the user has a surround system or stereo headphones he can determine where the

sound is coming from and then travel to that direction. Once the cluster has been found, the user can isolate that cluster (genre in this case) to play other similar tracks and then isolate further to play only songs by a given artist. There even is a doppler effect for the sounds in the environment so sounds you travel away from appear to be playing a little bit slower than those you travel to.

A 2D HUD system will tell the user which genres are the closest ones and which song is playing the loudest.

## Technology

All the rendering is done in the engine described in a short article below. FMOD is used for maintaining 3D sounds and all the technology behind that, e.g. the doppler effect, panning and volume control. I created a c++/python wrapper for it to work with the engine as a module. FMOD is being used today by many of the biggest game developers for large titles, including: **Guitar hero III, BioShock, Call of duty 4, Crysis, Starcraft II and World of Warcraft.**

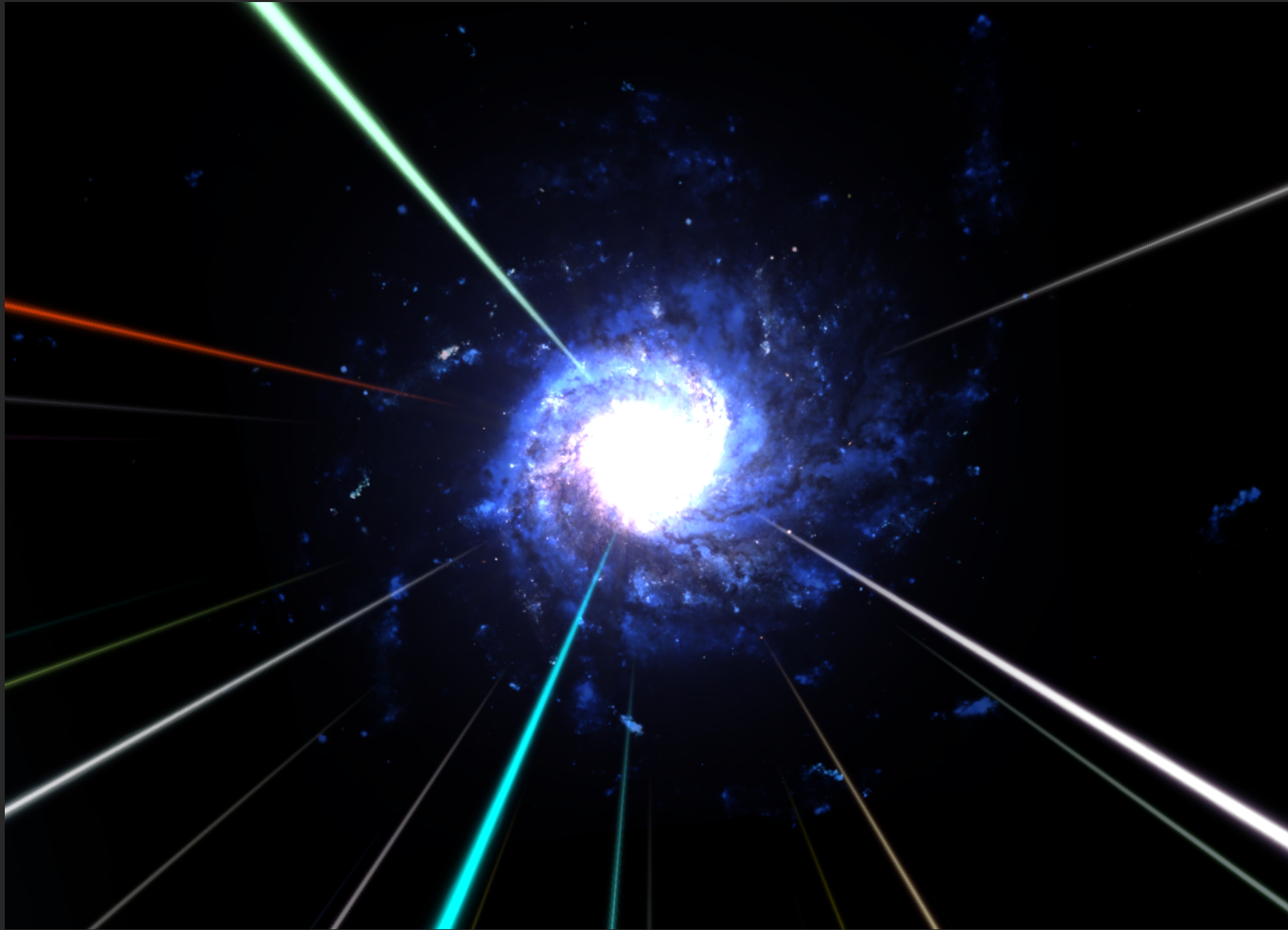
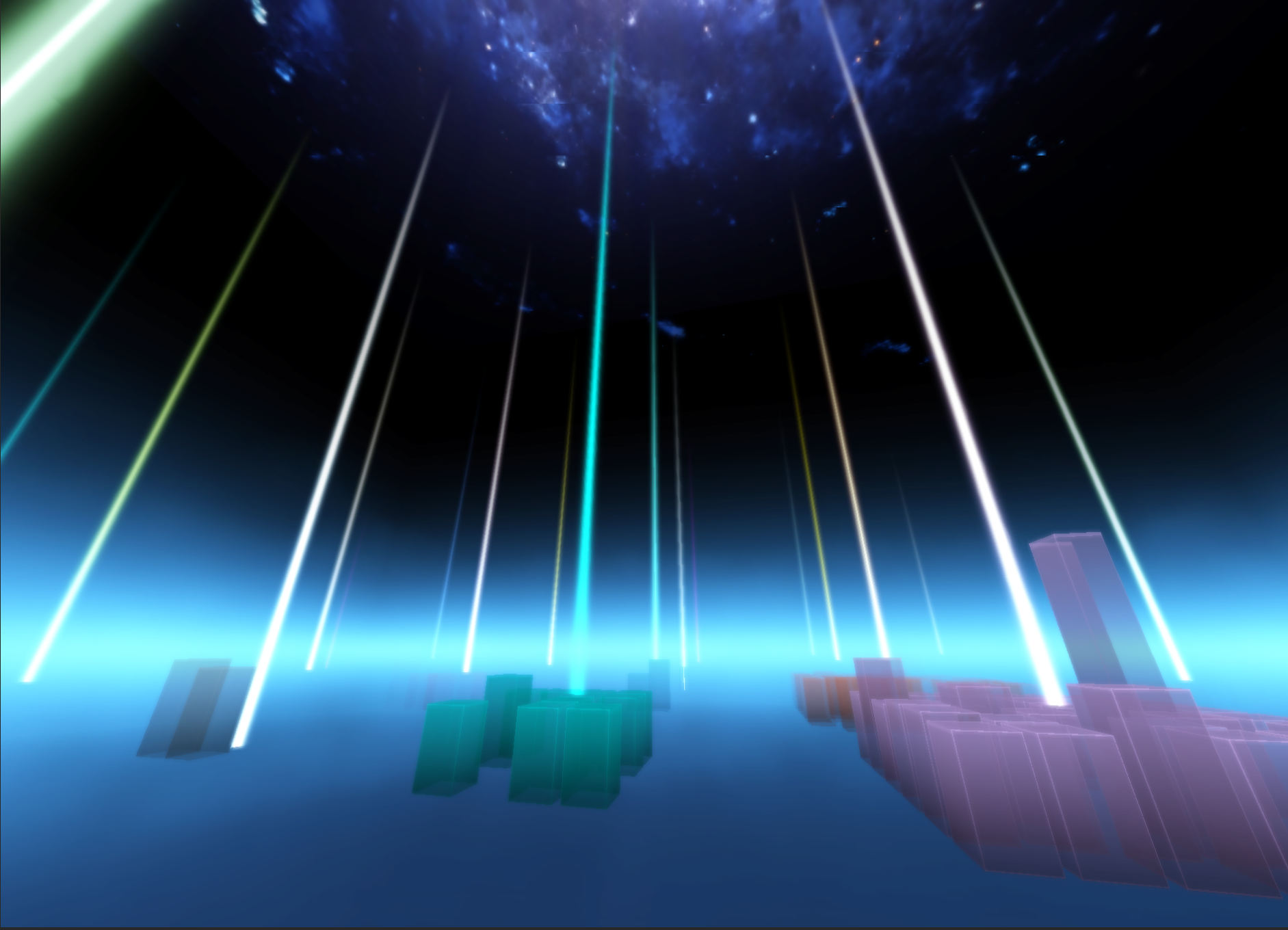
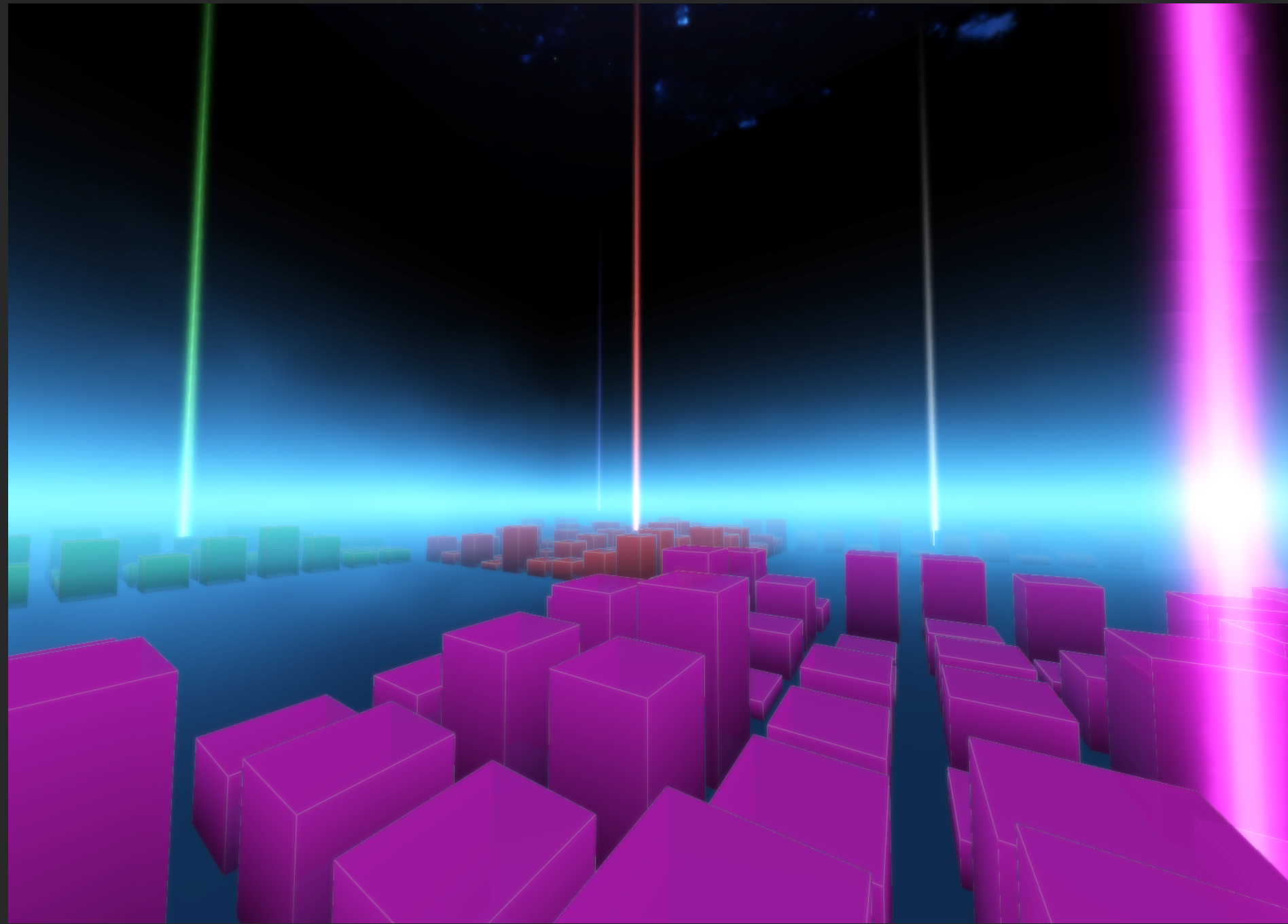
## Further developments

There is a lot of work to be done for this to grow from a research project to a full blown application. Further developments include:

- Improving stability and performance
- Build for MacOSX. Linux and Windows are already there
- Waveform and spectrum graphics for the tracks playing like all music-players have
- Portability for lower-end graphics cards, or not :)

author **Edvald Ingi Gislason**  
contact **edvaldig@gmail.com**

university **DTU (Technical University of Denmark)**  
department **IMM (Informatics and Mathematical Modelling)**  
course **Real-time graphics for virtual reality and games**  
semester **Spring, 2008**



# The Engine

The rendering engine is written in C++ and Python with the help of the Boost::Python libraries and the rendering it self is done using OpenGL.

*Python is a dynamic object-oriented programming language that can be used for many kinds of software development. It offers strong support for integration with other languages and tools, comes with extensive standard libraries, and can be learned in a few days. Many Python programmers report substantial productivity gains and feel the language encourages the development of higher quality, more maintainable code.*

-Python.org

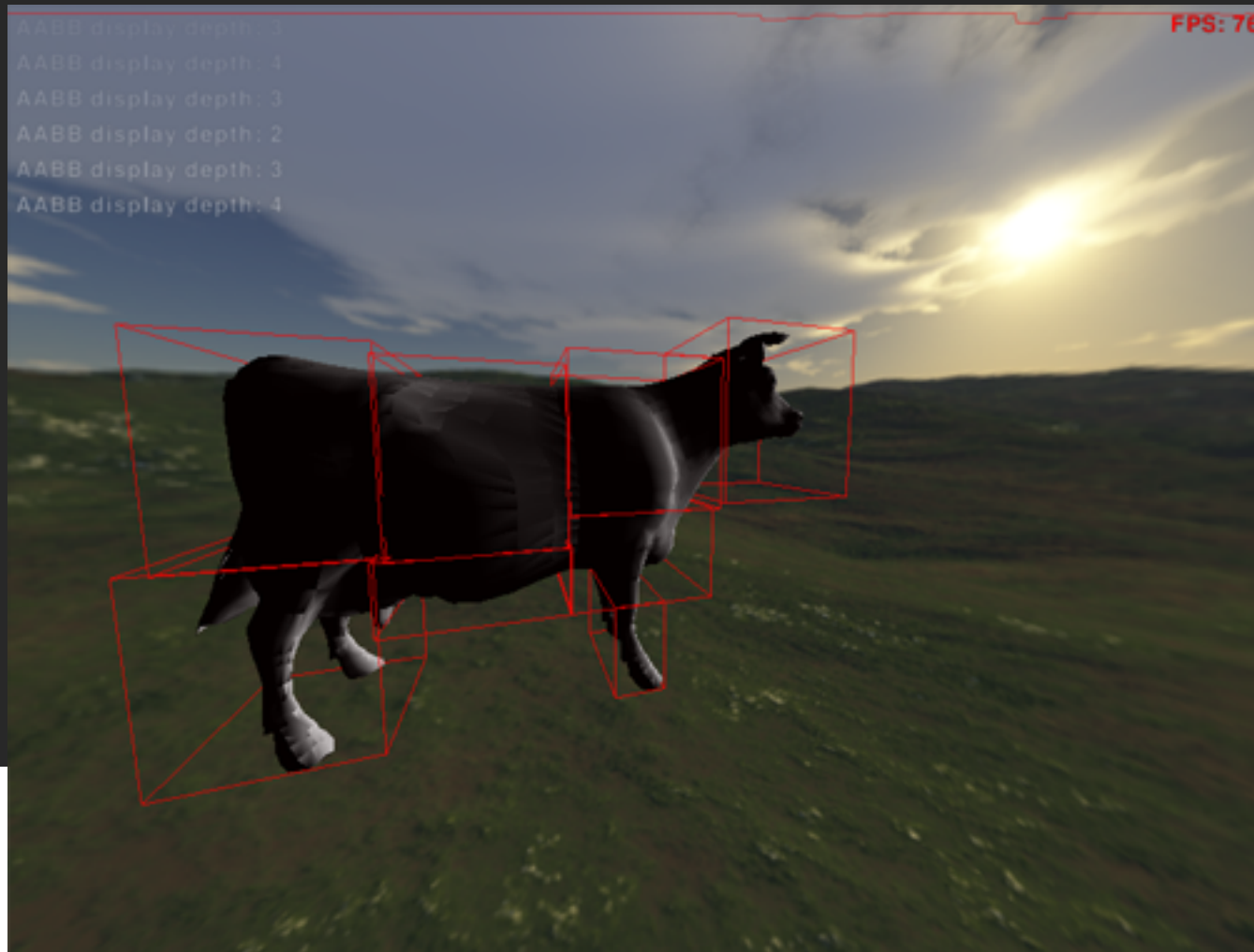
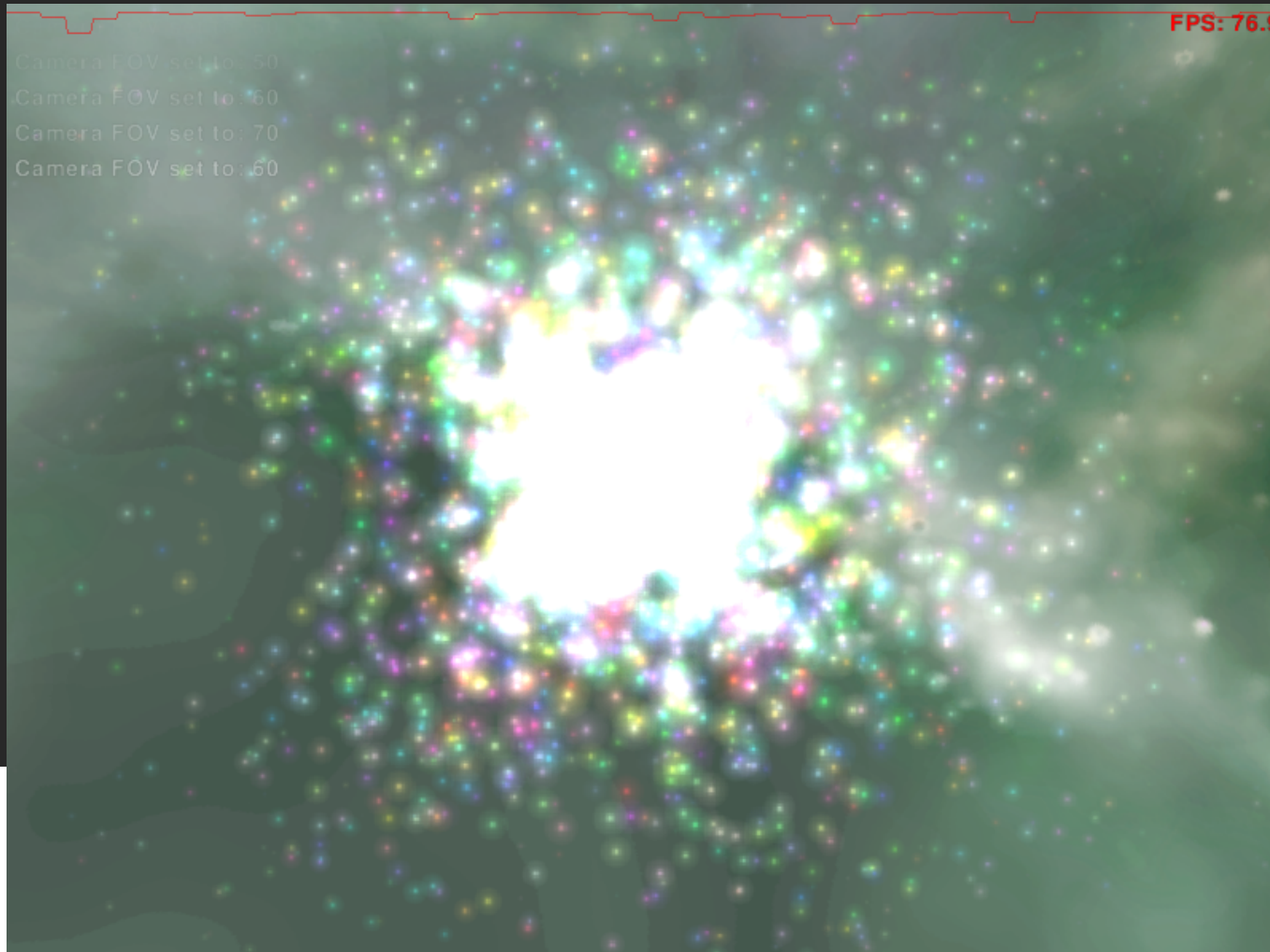
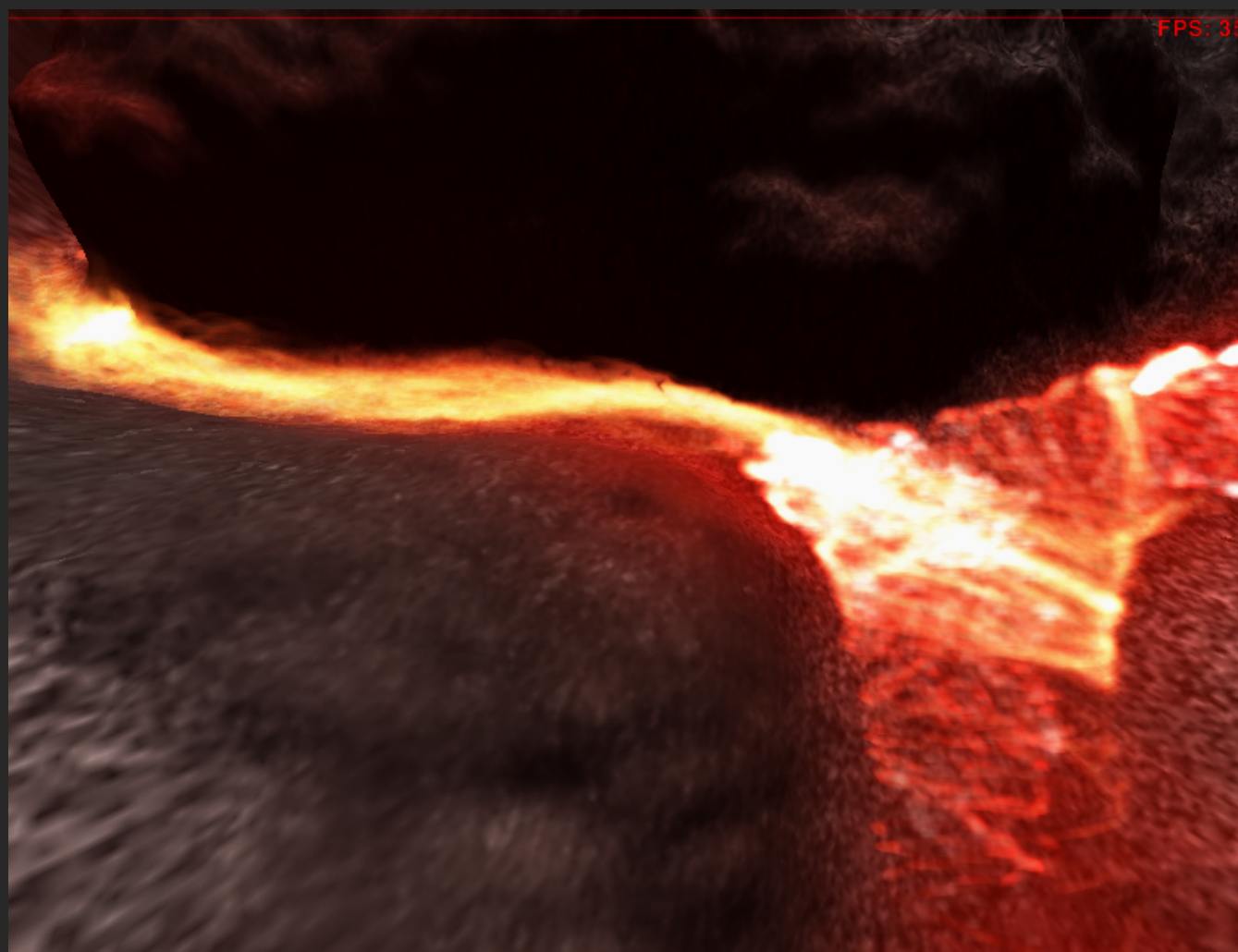
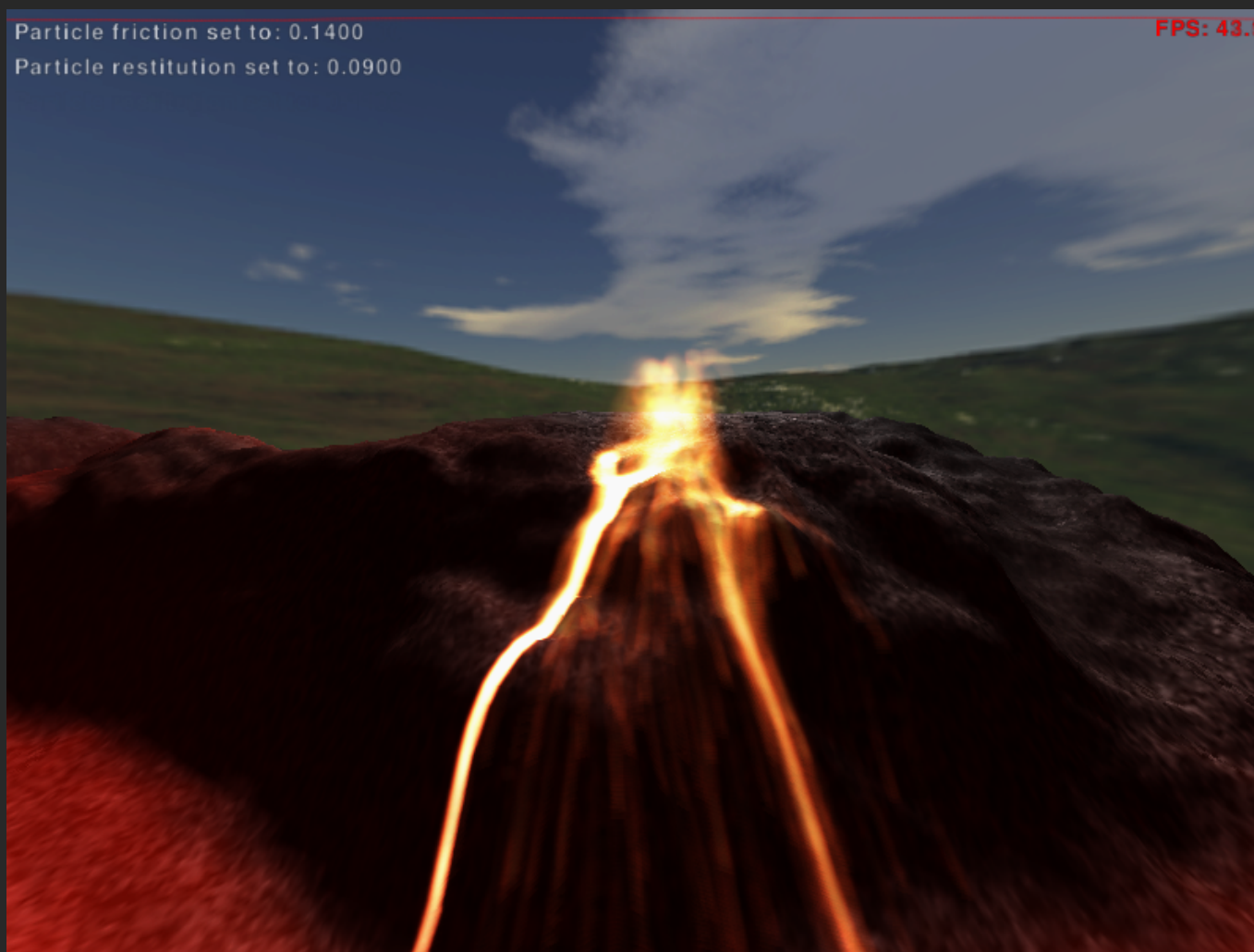
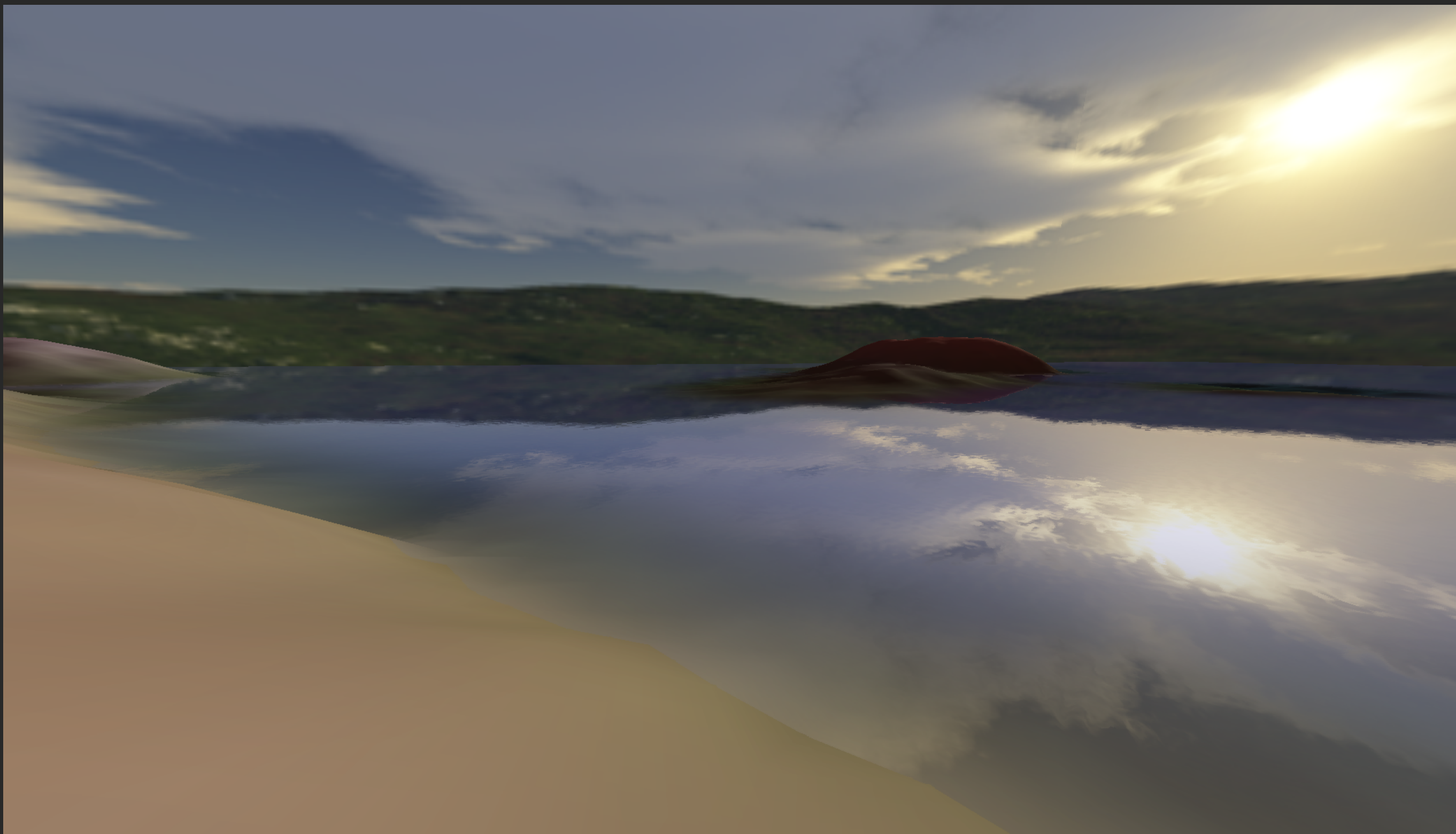
Demanding tasks, like large iterations and data manipulations are written in C++ but the main modules which bind everything together to form the framework are written in Python. This cooperation between the two languages offers the programmer a great platform to rapidly develop applications and games in much fewer lines of code compared to pure C++ implementations, without sacrificing too much performance. Large software development companies are using Python such as Google, Youtube.com, Industrial Light & Magic and CCP, an Icelandic gamedeveloper which is best known for it's game EVE-Online:

*"Python enabled us to create EVE Online, a massive multiplayer game, in record time. The EVE Online server cluster runs over 25,000 simultaneous players in a shared space simulation, most of which is created in Python. The flexibilities of Python have enabled us to quickly improve the game experience based on player feedback,"*  
*said Hilmar Veigar Petursson of CCP Games.*

-CCP Games

The engine is still in early development but it's features currently include:

- Resource management
  - Textures, Shaders, Meshes, Fonts, Sounds
- AABB Tree generation for meshes
- 3D Sound system (FMOD), sound emissions in a 3D environment
- Post processing using shaders, framebufferobjects and renderbuffers
- Frustrum culling with projection plane extraction
- Particlesystems using vertexbuffers, point sprites and shaders for maximum efficiency
- Console
- Basic 2D GUI System
- Module oriented, games and applications using the engine are not part of the engine itself
- SDL as a window manager provides portability to Unix, Linux, OSX, Windows and other platforms



powered