Progress Report – Voxel Populi:
A Decentralised Peer-to-Peer Voxel-Based World

Samuel J. Sully (sjs252)

26 January 2020

Project Supervisor: Prof. Jon Crowcroft
Director of Studies: Prof. Alan Mycroft
Project Overseers: Prof. Marcelo Fiore & Dr. Amanda Prorok

1   Status of Project

My project aims to create a 3D online world using a decentralised peer-to-peer network architecture. This involves 3 main components: a Kademlia distributed hash table (DHT), the game server and the client. I am approximately keeping up with my timetable and have more or less passed my success criteria. Though there is one item which I have not completed. The item in question is a small test agent to be used in evaluating the project which will be fairly simple to implement. This has not yet been needed and I've spent more time testing and debugging the rest of the project first before I move onto evaluation.

1.1   Work Completed

1. I have implemented the Kademlia DHT, based on the specification set out in the Kademlia paper [1]. I have made some adjustments and approximations to suit my project. For example, my implementation of Kademlia has two separate `STORE` and `FIND_VALUE` RPCs, one for player data and one for world data, this ensures these types of data are not conflated.

2. I have implemented the game server to run alongside the Kademlia network. The client uses the Kademlia DHT to locate the appropriate server for a particular section (chunk [2]) of the world, it can then connect to this server and download the relevant part of the world. The game server also handles simple updates such as player movement, changes in time and terrain editing by the player, these changes are stored on the server and multicast to other clients in (or nearby to) this chunk of the world.

3. I have produced a simple 3D client to interact with the world. I had initially proposed to develop this using LWJGL [3], however, in the end I decided to use Unity [4] to create the client as this was slightly simpler, and the graphical component is not the focus of this project. This section did take longer than I had anticipated, I spent quite a while working out the best way to generate the meshes for the terrain in such a way that the performance of the client was acceptable.

2   Work Remaining

I still need to implement my simple test client, which will be used to stress test the system. Additionally, further work on the game server to perform additional validity checking of player actions (such as ensuring they do not fall through terrain).

I will then start evaluation, this will be done by scaling up the system using AWS, as suggested in my project proposal. I will also be performing tests on single node performance, to see how many concurrent clients a single server can handle before it fails.

Finally, I will work on any extensions I have time for and do my write-up in parallel. I plan to have a near complete dissertation by the end of this term.

3   Extensions

In my original proposal I suggested several possible extensions, I have already implemented one of these (terrain generation using coherent noise). I plan to attempt to implement two further extensions, one of which was proposed in my initial proposal:

1. I propose to incorporate some simple redundancy into my system so that if a single node dies then the world data for that chunk of the world will not be regenerated from scratch by another node (meaning anything built there is lost).

2. I would also like to add some non-player entities to the world so that the game server is doing more, making this a more realistic simulation of an actual game server. So I will be adding some mobs [5] if I find time to complete this extension. This was suggested in my proposal.

4   Difficulties Faced

So far I have faced few difficulties, I would note that I unfortunately lost some time during the Christmas vacation due to illness. I feel that I compensated for this in the remainder of the holiday though. Additionally,

I might note that the Kademlia implementation took considerably longer than anticipated, mainly because of poor documentation and the fact that there are two different versions of the Kademlia paper which give conflicting specifications. I eventually discovered this discrepancy and settled on the implementation outlined in the earlier paper which was simpler and more clearly defined.

References

[1] Maymounkov, P. and Mazières, D. Kademlia: A Peer-to-peer Information System Based on the XOR Metric. `https://www.scs.stanford.edu/~dm/home/papers/kpos.pdf`. Accessed: 2020-01-26.

[2] "Chunk" on the Minecraft Wiki. `https://minecraft.gamepedia.com/Chunk`. Accessed: 2020-01-26.

[3] LWJGL: Lightweight Java Game Library. `https://www.lwjgl.org/`. Accessed: 2020-01-26.

[4] Unity Game Engine. `https://unity.com/`. Accessed 2020-01-26.

[5] "Mob" on Wikipedia. `https://en.wikipedia.org/wiki/Mob_(gaming)`. Accessed: 2020-01-26.