

Noobwerkz platform – Progress report:

The past two weeks have been spent learning a rather new system for graphics – BGFX – that allows an application to display graphics on anything that has a screen. It took a while to get adjusted to its way of doing things (two or three layers of abstraction and attendant toolkits), but I am proud to state that it now runs on both Android and desktop perfectly.

The loss of time from having had to replace a deeply-buried component has slowed down development by a week or more. However burdensome and unrewarding this effort was in the short-term, it will prove to be a great effort-saver in the long run, as BGFX contains working cross-platform implementations of a number of advanced graphics techniques. The time it took to integrate was a reflection of tricky design decisions that had to be made; most had to do with where this library fits in conceptually (deeper than what the programmer should usually see in day-to-day work) and if incorporating it would require a change of the entire build system (it didn't.)

Now, work is back on track almost precisely where it was two weeks ago, with the grand exception that whatever is written will now run on all popular hardware. This is a major win.

Work directly related to the game engine that needs to take place:

1. Redo 3D model loading code using BGFX instead of OpenGL – easy
2. Write a gradient mapping shader – fewer big files shunted around in memory (very costly) at the expense of a few more calculations on the video card (rather cheap) – medium difficulty
3. Terrain generation – medium difficulty
4. Terrain rendering (tri-planar shader + keeping triangle count low) – medium/advanced difficulty
5. Physics engine (re)integration – easy
6. Skeletal animation – medium
7. Pathfinding/navigation – medium/advanced difficulty
8. Sound – easy
9. Networking – medium/advanced difficulty

Also of note is the amount of learning with regards to web services – I have been reading up a lot on the pros and cons of various web server software and the niche in which each of those components belong. For a static, public-facing website I shall likely use a server called Nginx, which is easy to use and help to load pages quite quickly.