**The Problem, Objectives, Boundaries and Feasability**

*Seb Holzapfel*

**Name:** Metanact

**In a sentence**

**Metanact is a fast-paced, rogue-like, 2D action space-shooter; levels taking place within the client’s own file system.**

**The problem**

Increasingly, modern computer systems strive to hide the details of their operation, as not doing so is seen to be bombarding the user with extraneous information they do not require to complete tasks. This is not a phenomenon that is completely beneficial to society because, although computer users do not have to know of inner workings and technical details, it *is* important to understand the underlying complexity of their operations, perhaps to educate, or if only to sympathise more easily when, for example, their computer crashes.

Briefly, to my knowledge there is no educational, entertaining, highly interactive application which illustrates the complexity of (specifically) a computer’s file system. Hence, this is a *need* which my solution will attempt to fulfil.

**High-level objectives** (The solution should)

* Educate players somewhat as to the underlying structure of their filesystem
* Employ a sparse range of content to keep the player interested
* Provide intuitive game mechanics that can be easily familiarised with
* Operate in a stable manner on multiple operating systems

**Boundaries**

* **Time**, we are allocated a little more than 3 terms to complete the project
* **Lack of parallel labour** as this is not a team project
* **Performance requirements** are strict as in any game, (current aim is >40 FPS on an intel HD integrated graphics card)
* **Distributed program size** is aimed to be under 100MB.
* **In-game violence** should be kept to a minimum.

**The solution and it’s feasibility**

As described earlier, my solution will consist of an action space-shooter, whereby levels in-game correspond to and are influenced by directories on the player’s actual filesystem. Something like this is quite a large technical undertaking, especially when the entire game development team is of strength one. Begging the question – is it feasible? In response to the boundaries assigned:

* **Time-wise**, I have developed a similar application before. Not *quite* as complex of course, but it took about a month of on-off work to complete. Related to this as well, I wouldn’t be surprised if the application was in a working state in a couple of months – it’s polishing and content that takes the majority of time (especially in a game; bringing me to the next point).
* **A lack of parallel labour** means that I won’t just be writing code, I’ll be writing music, photoshopping sprites, recording sound effects, writing a campaign story, voice acting etc. I’ve got experience in each of these areas already though (maybe not the voice-acting), so this shouldn’t be a problem.
* It shouldn’t be too hard to stick to **performance requirements**, especially if the application is written in C++ (Not that I’ll be brute-forcing everything). My last C++ game ran at ~3000 FPS with a couple-hundred entities on screen and a decent computer.
* **Program size** shouldn’t be an issue as long as the game engine uses compressed assets, and doesn’t have too many dll dependencies.
* **Violence.** As long as campaign themes and visual style are kept clean, no small children should be corrupted in playing this game.