Metanact logo fades with the theme music which quietens as Seb talks

<Seb on nice background; all shot with DSLR - ‘Lead Programmer’ subtitled at bottom>

SEB: I guess you could say Metanact is a – um.. directory traversal game? Yeah; that works.

<Cut to game menu footage leading to in-game footage of destroying enemies; a few weapons etc. While footage is playing>

SEB: So the concept I wanted to explore with Metanact was the idea of having levels actually take place within the player’s filesystem. Things like folders being physical portals to other directories, enemies in the directory actually representing real files (with larger ones being tougher enemies); and some sort of convoluted story that could somehow lead to this strange scenario actually taking place.

<Cut through some various parts of cinematic>

<Cuts of gameplay again, filesystem images in background though>

SEB: As a 2D spaceshooter; throughout Metanact you’ll find ‘Scrap’; resources which will increase your score; enabling tougher enemies and exploration further into the depths of your filesystem, as well as the game’s story. Enemies will occasionally drop ‘Loot’, which enables you to customize your ship with new weapons, antigravity, and I’m looking at doing ship hulls & shields too if this happens to get off the ground. You’ll find the game tries to use the filesystem’s data as much as it can; by sticking your images in the background; randomly spitting text document’s text out -- that sort of thing. You’re basically -- Enacting on the Meta.

<Cuts to moving around directory, source files etc>

SEB: So, Metanact’s engine was written completely from scratch in C++. This was so I could get the game running as fast as possible because – reading directories on this scale generally isn’t quick – (unless you have an SSD but that’s another story). I’ve tried to make the game’s content as open and modifiable as I can – every game entity is defined in it’s own file where rules are set – like USE THIS IMAGE or COMBINE WITH THIS ENTITY or FOLLOW PLAYER or CREATES GRAVITY; that sort of stuff. If you want to create a NEW enemy; you just add it to the spawn-list file and it’ll get spawned on difficulty that you select. Same goes for loot, cinematics, weapons, subtitles, audio etc. Their use is all defined in files; outside of the game’s binary.

<Cut again to DSLR Seb>

SEB: But, where will your money actually go? Well; I’d say in its current state Metanact is only -- a little bit over halfway there. All the functionality exists; the problem hereon is basically CONTENT.

<Game footage>

More story, more weapons, more enemies, more types of loot, different ship types, proper voice-acting.

If the game gets funded, I’ll release the thing as-it-stands as soon as the funding period’s over. In the months afterwards; I’ll gradually introduce improvements and changes while listening to what you, the players, want. Hopefully; we can pipe out a finished version by the end of this year.

<DSLR Seb>

Well, Thanks for your support and I really look forward to seeing where we can take Metanact!