

## ET: Legacy Development - Feature #1093

### r\_worldLOD to set level-of-detail

05.05.2018 11:53 - keMoN

<b>Status:</b>	New	<b>% Done:</b>	0%
<b>Priority:</b>	Normal	<b>Spent time:</b>	0.00 hour
<b>Assignee:</b>			
<b>Category:</b>	Mod generic		
<b>Target version:</b>	2.78		
<b>OS:</b>		<b>Arch:</b>	

#### Description

I constantly think about level-of-details (LOD) for ET and how useful it would be to have. Now I thought of something that in my eyes **might** be possible, so hear me out:

We need a new brushmodel entity, something along the lines like func\_worldLOD.

This entity requires targetname and scriptname, but also has spawnflag choices: [0], [1] and [2].

Then we need a client cvar, something along the lines like r\_worldLOD. This cvar has the value [1] assigned by default.

[2] = highest level of details

[1] = regular level of details

[0] = lowest level of details

The mapper can then create the map like they normally would, starting with the basic geometry. Afterwards, when they start filling their basic rooms with details, they can differentiate between detail that, on the one hand, is there for gameplay purposes, for example to block line-of-sight and player movement, and on the other hand details that is out of the players movement area. Everything that is out of the players reach, for example brushes on roofs or brushes high up on walls, can all be assigned to a func\_worldLOD. func\_worldLODs with regular amount of detail receive spawnflag [1]. Everything that is truly excessive receives spawnflag [2].

Ingame, depending on the value of /r\_worldLOD, the appropriate amount of detail is rendered and the rest deleted. Changing the value would require a /vid\_restart to reload the map.

With this, mappers can finally create maps that are (**shameless self-promotion incoming**) around Dingskirchen's amount of detail. Everyone who wants to have an excessively detailed map can set /r\_worldLOD [2]. Everyone who is happy with the regular amount of detail can keep it on [1] and people that have troubles with even the regular amount of detail can set r\_worldLOD to [0].

- This is restricted to **brushmodels**, since misc\_models can't be assigned another entity class.
  - This should **only** be used on detail that is non-solid. Otherwise, this cvar is creating an unfair advantage.
  - Maps using this will **only** be playable on Legacy (presumably).
- ⇒ with great power comes great responsibility for mappers

I have attached screenshots, where I simulated what I'm talking about. Obviously, all three maps aren't excessively detailed, but with that cvar more details could be placed around the hallways of Railgun, etc. For example decals of water stains, rust or cracks in the concrete. More pipes, vents, toolboxes, etc. Everything highlighted in green is detail that is out of the player's reach and could potentially be assigned to a higher level of detail. Obviously, like I said, the materials used for them should be made non-solid via shaders, so people who decide to render them, don't have a disadvantage, because their nades get blocked.

Is this possible and if yes, is it good?

#### Files

r_worldLOD.jpg	413 KB	05.05.2018	keMoN
r_worldLOD2.jpg	490 KB	05.05.2018	keMoN
r_worldLOD3.jpg	494 KB	05.05.2018	keMoN