

Doomsday Engine - Bug #1126

MD2 model render pop

2013-07-25 09:28 - bcwood16

Status: New	Start date: 2013-07-25
Priority: Normal	% Done: 0%
Assignee:	
Category:	
Target version: Rendering	
Description MD2 models pop into view when centre is hidden by BSP Especially effecting long models such as custom pipes and beams. See attached example. Note - long models are supported with separate collision boxes (mobj) lined up along the model. This is due to the fact the collision box can only be 'box' shaped and making one large box to cover the long model is not practical. (as far as I know this should not effect MD2 renders) Labels: Renderer, Visual Glitch, 3D Models Attachments: <ul style="list-style-type: none">http://sourceforge.net/p/deng/bugs/1119/attachment/Model_pop.pk3	

History

#1 - 2013-07-25 10:20 - danij

- **Priority:** 1 --> 5

#2 - 2013-07-26 22:07 - skyjake

- **labels:** md2 pop --> Renderer, Visual Glitch, 3D Models

#3 - 2013-07-27 03:43 - skyjake

models.cpp:973:

```
// Half the distance from bottom left to top right.  
float radius = (def->scale[VX] * (max[VX] - min[VX]) +  
def->scale[VZ] * (max[VZ] - min[VZ])) / 3.5f;
```

The comment says "half", yet it's actually divided by 3.5. Changing it to 2 improves the situation but doesn't remove the popping completely.

Maybe there's a further issue with the contact spreading, or the radius isn't calculated correctly for this model?

#4 - 2013-08-05 22:41 - danij

Indeed this calculation should divide by two. I'm not sure where the 3.5 comes from. Perhaps it was originally intended as fudge factor...

I suspect the issue is either the radius calculation for the model. Or the relative position of the model vs the origin. Depending on the relative positions of the submodels, etc. The vertex coordinate ranges for each submodel should first be translated by any offset at DED level then united into a single bounding cuboid. Half the length of the diagonal of this cuboid should then yield the maximum spread range from the origin of the model, assuming it is centered on the origin (however if this is not the case then we would need to translate the origin of the spread as well).

Note that I've not looked at the model yet, or checked through the algorithm.

#5 - 2016-08-09 10:25 - skyjake

- *Target version deleted (1.8.6)*

#6 - 2017-04-03 14:41 - skyjake

- *Target version set to Rendering*