

Doomsday Engine - Feature #1945

Feature # 7 (Progressed): Next-gen renderer (codename "Gloom")

Efficient reuse of world geometry across multiple frames/passes

2015-01-13 13:44 - skyjake

<b>Status:</b>	Resolved	<b>Start date:</b>	2015-01-13
<b>Priority:</b>	Urgent	<b>% Done:</b>	100%
<b>Assignee:</b>	skyjake		
<b>Category:</b>	Redesign		
<b>Target version:</b>	3.0		
<b>Description</b> <p>One of the most fundamental performance problems of the old 1.x renderer is that every time the player view is drawn, world surfaces are recomputed into GL geometry. (One can see how costly this is by freezing the rendering lists.) However, many rendering techniques require rendering all or some parts of the world multiple times: VR modes draw dual views, shadow mapping requires passes from lights' point of view, reflections need several passes for dynamic cube maps, and overall most of the world remains static across frames so recomputing geometry is wasted effort.</p> <p>In practice, geometry should be stored as a reasonably small number of static vertex buffers. We should explore if moving planes could be partially or even completely implemented similarly to skeletal animation, where certain vertices would be affected by selected transformations. The trick would be to do this efficiently only for the needed planes, and only when the planes are expected to move. (After all, it is not necessary to efficiently support for <b>all</b> planes to move in the map with no foreknowledge.)</p> <p>Benchmarking:</p> <ul style="list-style-type: none"><li><a href="http://www.moddb.com/games/doom-compile-project">http://www.moddb.com/games/doom-compile-project</a></li></ul>			
<b>Related issues:</b>			
Related to Feature #1628: Surface meshes		<b>New</b>	<b>2013-10-22</b>
Related to Feature #1745: Performance optimization		<b>Progressed</b>	<b>2014-04-09</b>
Related to Feature #2136: Add a map rendering abstraction layer for supportin...		<b>Rejected</b>	<b>2015-11-28</b>
Related to Feature #1308: Splitscreen multiplayer		<b>Progressed</b>	<b>2003-11-01</b>

History

- #1 - 2015-01-13 13:45 - skyjake
  - Description updated
- #2 - 2015-01-13 13:50 - skyjake
  - Related to Feature #1628: Surface meshes added
- #3 - 2015-01-13 13:58 - skyjake
  - Tags changed from *Renderer, MapData, Performance* to *Renderer, MapData, Performance, OpenGL*
- #4 - 2015-01-13 14:00 - skyjake
  - Tags changed from *Renderer, MapData, Performance, OpenGL* to *Renderer, MapData, Performance, OpenGL, GL2*
- #5 - 2015-03-21 22:46 - skyjake
  - Related to Feature #1745: Performance optimization added
- #6 - 2016-11-28 09:35 - skyjake
  - Target version set to *Rendering*
- #7 - 2016-11-28 09:37 - skyjake
  - Related to Feature #2136: Add a map rendering abstraction layer for supporting both *Classic* and *Modern* rendering modes added
- #8 - 2017-12-05 14:53 - skyjake
  - Description updated

**#9 - 2018-10-27 15:08 - skyjake**

- *Status changed from New to Resolved*
- *Assignee set to skyjake*
- *Target version changed from Rendering to 3.0*
- *% Done changed from 0 to 100*

Gloom has an entirely different model for managing the map geometry.

For details, see the blog: [Further rendering explorations](#)

**#10 - 2019-11-29 16:28 - skyjake**

- *Related to Feature #1308: Splitscreen multiplayer added*