

# Modelling an I-Section in a 3D Graphics Library

Asked 16 years, 1 month ago    Modified 5 years, 10 months ago

Viewed 304 times

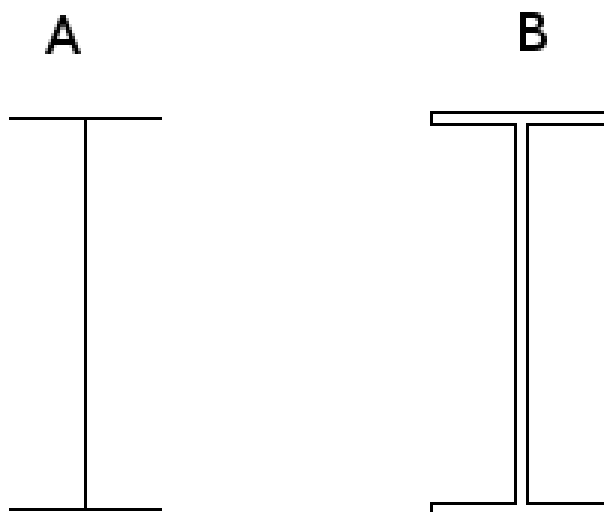


2

I am using Direct3D to display a number of I-sections used in steel construction. There could be hundreds of instances of these I-sections all over my scene.



I could do this two ways:



Using method A, I have fewer surfaces. However, with backface culling turned on, the surfaces will be visible from only one side. If backface culling is turned off, then the flanges (horizontal plates) and web (vertical plate) may be rendered in the wrong order.

Method B seems correct (and I could keep backface culling turned on), but in my model the thickness of plates

in the I-section is of no importance and I would like to avoid having to create a separate triangle strip for each side of the plates.

Is there a better solution? Is there a way to switch off backface culling for only certain calls of DrawIndexedPrimitives? I would also like a platform-neutral answer to this, if there is one.

graphics

directx

direct3d

3d

Share

Improve this question

Follow

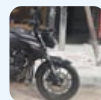
edited Feb 14, 2019 at 20:51



Glorfindel

22.6k ● 13 ● 89 ● 116

asked Oct 27, 2008 at 8:59



Agnel Kurian

59.4k ● 47 ● 153 ● 225

4 Answers

Sorted by:

Highest score (default)



1



First off, backface culling doesn't have anything to do with the order in which objects are rendered. Other than that, I'd go for approach B for no particular reason other than that it'll probably look better. Also this object probably isn't more than a hand full of triangles; having hundreds in a scene shouldn't be an issue. If it is, try looking into hardware instancing.



Share Improve this answer

answered Oct 27, 2008 at 9:13

Follow



**Jasper Bekkers**

6,809 ● 36 ● 46



In OpenGL you can switch of backface culling for each triangle you draw:

1



```
glEnable(GL_CULL_FACE);  
glCullFace(GL_FRONT);  
// or  
glCullFace(GL_BACK);
```



I think something similar is also possible in Direct3D

Share Improve this answer

answered Oct 27, 2008 at 9:13

Follow



**Peter Parker**

29.7k ● 5 ● 54 ● 81

Notice that enabling and disabling backface culling for each triangle separately will be a huge performance drain because it's a serverside state. – [Jasper Bekkers](#) Oct 27, 2008 at 9:16

It is possible in Direct3d with `IDirect3DDevice9::SetRenderState()` call, though as Jaspers pointed out, it is a big hit on performance switching that for each mesh. One could turn off culling once and render all the I-sections at once, two switches shouldn't be a burden.

– [Saulius Žemaitaitis](#) Jan 24, 2009 at 11:30



If your I-sections don't change that often, load all the sections into one big vertex/index buffer and draw them

1



with a single call. That's the most performant way to draw things, and the graphic card will do a fast job even if you push half a million triangle to it.



Yes, this requires that you duplicate the vertex data for all sections, but that's how D3D9 is intended to be used.



Share Improve this answer

answered Oct 27, 2008 at 9:32

Follow



[Nils Pipenbrinck](#)

86.2k ● 33 ● 155 ● 223



1

I would go with A as the distance you would be seeing the B from would be a waste of processing power to draw all those degenerate triangles.



Also I would simply fire them at a z-buffer and allow that to sort it all out.



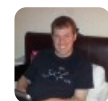
If it get's too slow then I would start looking at optimizing, but even consumer graphics cards can draw millions of polygons per second.



Share Improve this answer

answered Oct 27, 2008 at 13:35

Follow



[graham.reeds](#)

16.5k ● 17 ● 75 ● 138