

## VisIt - Bug # 1230: Find shader solution for rendering nodal values on quads

<b>Status:</b>	Pending	<b>Priority:</b>	Normal
<b>Author:</b>	Cyrus Harrison	<b>Category:</b>	
<b>Created:</b>	11/06/2012	<b>Assigned to:</b>	
<b>Updated:</b>	11/28/2012	<b>Due date:</b>	
<b>Likelihood:</b>	3 - Occasional		
<b>Severity:</b>	3 - Major Irritation		
<b>Found in Version:</b>	2.5.2		
<b>OS:</b>	All		
<b>Support Group:</b>	Any		
<b>Subject:</b>	Find shader solution for rendering nodal values on quads		
<b>Description:</b>	<p>We need a solution for the long standing anisotropic rendering of nodal vars on quads.</p> <p>This happens b/c quads are split into two triangles by OpenGL &amp; the bi-linear interpolation we want isn't realized.</p> <p>There have been a few attempts to resolve this:</p> <ul style="list-style-type: none"><li>*Subdivide quads operator It doesn't avoid the fundamental splitting issue, and its defaults are targeted a very large quads. This makes it very hard for users to try.</li><li>*Splitting to four triangles Still isn't high enough fidelity, for a symmetric dataset, can make everything look hashy.</li></ul> <p>I think the only solution is a shader strategy that splits quads into two tris, but provides all of the quad info so the proper interpolation can be done. This should be prototyped outside of VTK. Adding it to VTK is going to be quite difficult :-).</p> <p>Here is some more info: <a href="http://visitusers.org/index.php?title=Pseudocolor_rendering_of_nodal_variables_on_quads">http://visitusers.org/index.php?title=Pseudocolor_rendering_of_nodal_variables_on_quads</a></p>		

### History

11/28/2012 11:17 am - Eric Brugger

- Status changed from New to Pending