Subject: Re: dxf (dwg) read & write ?
Posted by Karl Schultz on Wed, 15 Mar 2006 16:02:12 GMT
View Forum Message <> Reply to Message

I'm not a DXF expert, but I can tell you that the IDL code is just issuing
a warning here. It is going ahead and writing your vertex list out to
the DXF Polyline, without paying any attention to the connectivity list.
The connectivity member is there in the IDL_DXF_POLYLINE structure mainly
for reading in polylines and polyline meshes. You can really only *write*
simple polylines consisting of a list of vertices. In this respect, the
implementation is a little unbalanced because it offers more reading
functionality than writing functionality.

If you have multiple lines in your IDL Polyline, you'll probably have to
extract each line out into its own vertex list and make individual DXF
Polyline Entities for each one.

As far as closing the polyline goes, I would simply try adding a final
vertex that is the same as the first vertex in the vertex list. There
appears to be no code in the PutEntity path to set the closed flag in the
DXF entity. Again, the GetEntity code checks this flag in the DXF entity
and if it is on, adds the index of the first vertex onto the end of the
connectivity list.

Karl

On
Tue, 14 Mar 2006 17:45:45 -0800, b_gom wrote:

> I seem to only be able to find references to reading DXF files. Does
> anyone out there have a more sophisticated example of -writing- DXF
> files than the one in the help documents? In particular, it would be
> nice to know how to write out closed polylines; I always get an error
> "IDLFFDXF::PUTENTITY: DXF error: DXF Polyline: explicit connectivity
> ignored" whenever I try to pass a connectivity array to the polyline
> object type.
>
> Brad
>
>
>
> Rick Towler wrote:
>> news reader wrote:
>>> with idl 6.2 ?
>>
>> DXF, Yes, mostly, sort-of. DWG, no.
>>
>> Check out the docs on IDLffDXF. I've put together an object that

>> simplifies reading DXF files, handling the more mundane tasks of atom
>> creation and cleanup. The one big limitation is that I don't have any
>> dxf files that have any meaningful color information so while there is
>> some code in place to handle color, it is untested. I end up brute
>> forcing the color by getting a list of the primitives after loading the
>> file and coloring them individually via their SetProperty method.
>>
>>
>> http://www.acoustics.washington.edu/~towler/programs/rhtgrdx fmodel__define.pro
>>
>> -Rick
