
Subject: Re: Vector output of idlgrpolygon models
Posted by [David Fanning](#) on Tue, 08 Nov 2011 14:48:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

D D writes:

> iii) A better question may be how to fix the pdf output/is this a
> known issue?

Since this is a one-off, have you tried just creating
a large PostScript file and converting that to PDF with,
say, Adobe Distiller, where you have a lot of control
over the parameters?

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Vector output of idlgrpolygon models
Posted by [D D](#) on Tue, 08 Nov 2011 15:43:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Nov 8, 2:48 pm, David Fanning <n...@dfanning.com> wrote:

> D D writes:
>> iii) A better question may be how to fix the pdf output/is this a
>> known issue?
>
> Since this is a one-off, have you tried just creating
> a large PostScript file and converting that to PDF with,
> say, Adobe Distiller, where you have a lot of control
> over the parameters?
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>

> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

This is one thing i've not tried yet. I've not got access to Distiller but I guess the idea of using IDL to export to ps and then using an external program to print to a pdf printer may work. Speaking of which I could probably try the idl demo on a windows machine with a decent pdf print driver and then use the idlgrprinter directly.

Thanks for the suggestion.

David

Subject: Re: Vector output of idlgrpolygon models
Posted by [David Fanning](#) on Tue, 08 Nov 2011 15:54:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

D D writes:

> This is one thing i've not tried yet. I've not got access to Distiller
> but I guess the idea of using IDL to export to ps and then using an
> external program to print to a pdf printer may work. Speaking of which
> I could probably try the idl demo on a windows machine with a decent
> pdf print driver and then use the idlgrprinter directly.

There are many PS to PDF converters that are a LOT less expensive than Adobe Distiller. Some of them work well, and some not so much. Distiller is VERY reliable, which is why I use it. But, something else may work as well.

Have you tried using Dialog_PrinterSetup on your machine to select a different output printer? I don't remember, off-hand, how to set up a printer on a UNIX machine, except that I remember it was a pain in the neck. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Vector output of idlgrpolygon models
Posted by [D D](#) on Tue, 08 Nov 2011 16:13:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Nov 8, 3:54 pm, David Fanning <n...@dfanning.com> wrote:

> D D writes:
>> This is one thing i've not tried yet. I've not got access to Distiller
>> but I guess the idea of using IDL to export to ps and then using an
>> external program to print to a pdf printer may work. Speaking of which
>> I could probably try the idl demo on a windows machine with a decent
>> pdf print driver and then use the idlgrprinter directly.
>
> There are many PS to PDF converters that are a LOT less
> expensive than Adobe Distiller. Some of them work well,
> and some not so much. Distiller is VERY reliable, which
> is why I use it. But, something else may work as well.
>
> Have you tried using Dialog_PrinterSetup on your machine
> to select a different output printer? I don't remember,
> off-hand, how to set up a printer on a UNIX machine,
> except that I remember it was a pain in the neck. :-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

I have a licence for CorelDraw which I know can import postscripts and
I think can publish to pdf so this will be my first target I think.

I've played around with Dialog_PrinterSetup but couldn't find any way
to really tweak anything useful, there seem to be two main options
(xprinter and a HP printer, which is the same across several machines
on different sites). There is an option to add a new printer but it
doesn't seem to pick up on any printers installed on the system and
just asks you to pick from a large list of predefined options. After
picking one it doesn't seem to actually allow you to add the printer
(although i'm probably just not doing it right).

Having checked the machines which have pdf support I noticed that my
Draw widget is using software rendering. Could this be the cause for
the "missing" polys? I recall an old message from this list where
there was a problem with the draw order when using software rendering,

so this is something I will check (though this is academic as I have no way of using idl 8 on a machine which does have hardware enabled).

Thanks,
David

Subject: Re: Vector output of idlgrpolygon models
Posted by [David Fanning](#) on Tue, 08 Nov 2011 17:12:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

D D writes:

- > Having checked the machines which have pdf support I noticed that my
- > Draw widget is using software rendering. Could this be the cause for
- > the "missing" polys? I recall an old message from this list where
- > there was a problem with the draw order when using software rendering,
- > so this is something I will check (though this is academic as I have
- > no way of using idl 8 on a machine which does have hardware enabled).

I don't remember any problem with draw order using software rendering. I would be more inclined to think that you may have that backwards. Generally speaking, we turn to software rendering when everything else is hopeless. (Not to say there couldn't be problems in software rendering, too. It wouldn't surprise me much. Just more than I would be surprised to find problems in hardware rendering.)

Hardware and software rendering can be turned on for the individual draw widget or window, so I'm guessing if you are running the program, you probably have some kind of control.

You may not have a machine with a decent graphics card. But that's something else, entirely. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Vector output of idlgrpolygon models
Posted by [Karl\[1\]](#) on Tue, 08 Nov 2011 17:13:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

You might try looking at the REJECT property in IDLgrPolygon. If your surfaces are defined correctly so that the normals are correct, REJECT can be set to prevent drawing the polygons that face away from the viewer. This might reduce the number of unwanted polygons you are dealing with.

Also might look at the VECT_SORTING keyword in IDLgrClipboard::Draw(). Graphics hardware uses Z-buffers to take care of the hidden surface removal problem. The clipboard doesn't have such hardware and does coarse-grained sorting of the objects based on their depth instead. I can't remember if it uses the average Z of the entire IDLgrPolygon object, or the average Z of each triangle making up the polygon. If the latter is true, then that level of resolution may be good enough to sort out your surfaces. There will likely be problems where the toroids intersect, so look carefully there. The clipboard object won't split up intersecting triangles and draw just the visible pieces.

A more robust solution would be to use a BSP tree to sort them out and take care of splitting intersecting faces, but that is a lot of work.

Subject: Re: Vector output of idlgrpolygon models
Posted by [D D](#) on Tue, 08 Nov 2011 19:58:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Nov 8, 5:13 pm, Karl <Karl.W.Schu...@gmail.com> wrote:

> You might try looking at the REJECT property in IDLgrPolygon. If your surfaces are defined correctly so that the normals are correct, REJECT can be set to prevent drawing the polygons that face away from the viewer. This might reduce the number of unwanted polygons you are dealing with.

>

> Also might look at the VECT_SORTING keyword in IDLgrClipboard::Draw(). Graphics hardware uses Z-buffers to take care of the hidden surface removal problem. The clipboard doesn't have such hardware and does coarse-grained sorting of the objects based on their depth instead. I can't remember if it uses the average Z of the entire IDLgrPolygon object, or the average Z of each triangle making up the polygon. If the latter is true, then that level of resolution may be good enough to sort out your surfaces. There will likely be problems where the toroids intersect, so look carefully there. The clipboard object won't split up intersecting triangles and draw just the visible pieces.

>

> A more robust solution would be to use a BSP tree to sort them out and take care of splitting intersecting faces, but that is a lot of work.

I activate hardware rendering by default on my draw widgets, IDL then sets them to software if the machine doesn't have suitable hardware. Indeed I would have thought that the hardware to file comparison would be the most likely place for a difference to occur.

As most of my surfaces aren't closed setting the REJECT keyword to anything other than 0 means I lose half of my surface. Additionally many of the "hidden" polygons will have normals pointing towards (or away from) the camera. It's a shame because at first I thought REJECT was what I was after. It is an option if I enforce a fixed viewpoint but this is not great.

VECT_SORTING again looks useful but only alters the order items are drawn, not if they are drawn.

Currently I draw my surfaces as a single polygon object, I have tested splitting each surface into the individual polygons and redrawing. This makes the visualisation and manipulation a fair bit slower but still doesn't allow automated hidden object removal with vect_sorting or reject.

One (probably idiotic) thought i've just had is if it's possible for a user to click in the window to select a polygon (i.e. IDL can take an x-y position and tell you what you've clicked on) then you could technically click every polygon you can see and mark it, then once you've covered the screen toggle all the unmarked polygons HIDE property. How you would go about this in practice i'm not sure, moreover I'd guess this is likely to take a considerable amount of time if one were to scan over the full window at the smallest resolution. I'm sure some optimisation could be made but this still sounds like it's probably not a good way to go (if it's even possible).

Any other suggestions would be much appreciated but I think it looks like there's no simple way to achieve this. It will either involve writing a routine, putting up with large files or invoking an external tool (i'm trying hidden object removal with CorelDraw at the moment but it doesn't like the number of objects I think).

Thanks,
David

(Going slightly off topic [well off IDL anyway] : Searching for a solution to this I came across the C library GL2PS which apparently does this hidden object removal and is a routine for creating Postscript files from opengl commands. As I have little knowledge of opengl and no knowledge of C i've not got anywhere with it but it would certainly could form a nice output object! There is optional support for it in Paraview [which can read vrml] so this is one possible route.)

Subject: Re: Vector output of idlgrpolygon models
Posted by [penteado](#) on Mon, 21 Nov 2011 22:30:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Nov 8, 3:12 pm, David Fanning <n...@dfanning.com> wrote:
> I don't remember any problem with draw order using software
> rendering. I would be more inclined to think that you may have
> that backwards. Generally speaking, we turn to software rendering
> when everything else is hopeless. (Not to say there couldn't be
> problems in software rendering, too. It wouldn't surprise me much.
> Just more than I would be surprised to find problems in hardware
> rendering.)

I do remember an article mentioning some problems with hardware rendering:

http://www.idlcoyote.com/ng_tips/render.php

There was a problem with PS drawing order, logged years ago as CR ID 51003.
