
Subject: Axes etc. in XVOLUME

Posted by [K. Bowman](#) on Wed, 18 Feb 2004 18:40:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

I need to do some volume visualization, so I have been playing with XVOLUME.

Like many built-in IDL routines, XVOLUME assumes that the user is happy with "coordinate axes" whose units are array indices. This makes it useful for producing art, but not scientific graphs. I would like to provide coordinate values and names (not just "X"). It would also be nice to know just which isosurface is being rendered.

Has anyone made the necessary modifications to XVOLUME to do either of these things? I have looked at the XVOLUME code and rapidly found myself lost among the objects.

BTW, what does the 'Select' button do? I can select something in the plot, but then what?

I briefly tried iVolume, but I rapidly ended up with all of the buttons grayed out, unable to do anything. That program has me completely baffled.

Ken Bowman

Subject: Re: Axes etc. in XVOLUME

Posted by [aardvark62](#) on Thu, 26 Feb 2004 22:36:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Kenneth,

I put a Visio diagram on my webpage today. (www.paulsorenson.com) The diagram outlines IDLex classes, which underlie XVOLUME. The diagram is under the link entitled "XOBJVIEW:Under The Hood." That might be of some help if/when you are looking at the XVOLUME code.

-Paul Sorenson

Kenneth Bowman <k-bowman@null.tamu.edu> wrote in message news:<k-bowman-D91D8E.11402118022004@news.tamu.edu>...

> I need to do some volume visualization, so I have been playing with
> XVOLUME.

>

> Like many built-in IDL routines, XVOLUME assumes that the user is happy
> with "coordinate axes" whose units are array indices. This makes it
> useful for producing art, but not scientific graphs. I would like to

> provide coordinate values and names (not just "X"). It would also be
> nice to know just which isosurface is being rendered.
>
> Has anyone made the necessary modifications to XVOLUME to do either of
> these things? I have looked at the XVOLUME code and rapidly found
> myself lost among the objects.
>
> BTW, what does the 'Select' button do? I can select something in the
> plot, but then what?
>
> I briefly tried iVolume, but I rapidly ended up with all of the buttons
> grayed out, unable to do anything. That program has me completely
> baffled.
>
> Ken Bowman

Subject: Re: Axes etc. in XVOLUME

Posted by [Kenneth P. Bowman](#) on Fri, 27 Feb 2004 01:38:33 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article <8270ac8d.0402261436.5a6b5acc@posting.google.com>,
aardvark62@msn.com (Paul Sorenson) wrote:

> I put a Visio diagram on my webpage today. (www.paulsorenson.com)
> The diagram outlines IDLex classes, which underlie XVOLUME. The
> diagram is under the link entitled "XOBJVIEW:Under The Hood." That
> might be of some help if/when you are looking at the XVOLUME code.
>
> -Paul Sorenson

Thanks, but I don't have a Windows machine, so I cannot view the file.

Ken

Subject: Re: Axes etc. in XVOLUME

Posted by [David Fanning](#) on Fri, 27 Feb 2004 02:20:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

Kenneth P. Bowman writes:

> Thanks, but I don't have a Windows machine, so I cannot view the file.

I originally thought this was a 5 minute job. I found the
place where the code needed to be changed (in IDLexVolView__Define),
but when I made the simple change the volume disappeared from

the display! (Ah, well, object graphics after all.) Then I realized I was already 15 minutes into it and I could hear the big sucking noise already, so I backed off.

Something odd is going on in the NORMALIZE method of that object. Perhaps Paul will explain. But that's where I would look for things to fix.

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Axes etc. in XVOLUME
Posted by [Paul Sorenson](#) on Wed, 12 May 2004 23:07:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

I put a non-Windows version of the diagram on my website.
www.paulsorenson.com/underthehood.html .-Paul Sorenson>In article
8270ac8d.0402261436.5a6b5acc@posting.google.com>,
> aardvark62@msn.com (Paul Sorenson) wrote:

>> I put a Visio diagram on my webpage today. (www.paulsorenson.com)
>> The diagram outlines IDLex classes, which underlie XVOLUME. The
>> diagram is under the link entitled "XOBJVIEW:Under The Hood." That
>> might be of some help if/when you are looking at the XVOLUME code.
>>
>> -Paul Sorenson

> Thanks, but I don't have a Windows machine, so I cannot view the file.

> Ken

Subject: Re: Axes etc. in XVOLUME
Posted by [Paul Sorenson](#) on Fri, 21 May 2004 20:45:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

The NORMALIZE method has an optional keyword, ADJUST_AXES, that XVOLUME uses internally to mess with the axes. (The actual call to NORMALIZE is in `idlexvolviewwid__define.pro`.) With that keyword, axes are automatically adjusted to fit the rest of the graphics objects in the view. The

IDLexVolView has several non-axes graphics objects in it such as contour lines, a polygonal surface, the actual volume itself, etc. If we move all of these graphics objects, say, 10 units in x, the axes will automatically adjust themselves to span that range. Here is an example moving the volume 10 units by just hardcoding it. A similar technique would have to be applied to all the other non-axes graphics objects in IDLexVolView. Maybe that could be done in the SetProperty method when the volume property is set.

```
oVol = obj_new('IDLgrVolume', $
  keyword_set(test) ? $
    congrid(bytscl(randomu((seed=0), 4, 4, 4)), 40, 40, 20) $
    : vol, $
  /zbuff, $
  interpolate=interpolate, $
  hints=2, $
  /no_copy, $
  xcoord_conv=[10,1], $ ;move volume 10 units in x.
  /zero_opacity_skip $
)
```

There are examples using the ADJUST_AXES keyword on my webpage at <http://www.paulsorenson.com/underthehood.html> .

-Paul Sorenson
www.paulsorenson.com

> Kenneth P. Bowman writes:

>> Thanks, but I don't have a Windows machine, so I cannot view the file.

> I originally thought this was a 5 minute job. I found the
> place where the code needed to be changed (in IDLexVolView__Define),
> but when I made the simple change the volume disappeared from
> the display! (Ah, well, object graphics after all.) Then I
> realized I was already 15 minutes into it and I could hear
> the big sucking noise already, so I backed off.

> Something odd is going on in the NORMALIZE method of that
> object. Perhaps Paul will explain. But that's where I would
> look for things to fix.

> Cheers,

> David

> --

> David Fanning, Ph.D.

> Fanning Software Consulting
> Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Axes etc. in XVOLUME

Posted by [Paul Sorenson](#) on Sat, 22 May 2004 18:46:52 GMT

[View Forum Message](#) <> [Reply to Message](#)

Oops, I forgot to mention: the "oVol = obj_new('IDLgrVolume', \$..." code snippet is in file xvolume.pro.

-Paul Sorenson
www.paulsorenson.com

"Paul Sorenson" <aardvark62@msn.com> wrote in message
news:40ae6a52_3@127.0.0.1...

> The NORMALIZE method has an optional keyword, ADJUST_AXES, that XVOLUME
uses

> internally to mess with the axes. (The actual call to NORMALIZE is in
> idlexvolviewwid__define.pro.) With that keyword, axes are automatically
> adjusted to fit the rest of the graphics objects in the view. The
> IDLexVolView has several non-axes graphics objects in it such as contour
> lines, a polygonal surface, the actual volume itself, etc. If we move
all

> of these graphics objects, say, 10 units in x, the axes will automatically
> adjust themselves to span that range. Here is an example moving the
volume

> 10 units by just hardcoding it. A similar technique would have to be
> applied to all the other non-axes graphics objects in IDLexVolView. Maybe
> that could be done in the SetProperty method when the volume property is
> set.

>
> oVol = obj_new('IDLgrVolume', \$
> keyword_set(test) ? \$
> congrid(bytscl(randomu((seed=0), 4, 4, 4)), 40, 40, 20) \$
> : vol, \$
> /zbuff, \$
> interpolate=interpolate, \$
> hints=2, \$
> /no_copy, \$
> xcoord_conv=[10,1], \$;move volume 10 units in x.
> /zero_opacity_skip \$
>)
>

> There are examples using the ADJUST_AXES keyword on my webpage at
> <http://www.paulsorenson.com/underthehood.html> .

>
> -Paul Sorenson

> www.paulsorenson.com
>
>> Kenneth P. Bowman writes:
>
>>> Thanks, but I don't have a Windows machine, so I cannot view the file.
>
>> I originally thought this was a 5 minute job. I found the
>> place where the code needed to be changed (in IDLexVolView__Define),
>> but when I made the simple change the volume disappeared from
>> the display! (Ah, well, object graphics after all.) Then I
>> realized I was already 15 minutes into it and I could hear
>> the big sucking noise already, so I backed off.
>
>> Something odd is going on in the NORMALIZE method of that
>> object. Perhaps Paul will explain. But that's where I would
>> look for things to fix.
>
>> Cheers,
>
>> David
>
>> --
>> David Fanning, Ph.D.
>> Fanning Software Consulting
>> Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
>
>
