Subject: Re: how to ensure overwriting of eps graphics files?
Posted by David Fanning on Sun, 15 Apr 2007 20:13:00 GMT
View Forum Message <> Reply to Message

swingnut@gmail.com writes:

> Ah, the never-ending challenges of working in IDL. I just discovered a
> little while ago, while continuing to develop code for my
> dissertation, that IDL will write a new postscript file if none
> exists. Great. But in the process of debugging, setting reasonable
> ranges for plotting, adding and removing items from various plots,
> etc, I've stumbled upon an odd behavior whereby IDL is not overwriting
> existing eps files with the updated plots. (I discovered this because
> I changed something that altered which of the plots gets overwritten
> based on the analysis.)

I doubt this very, very much. I have never known IDL
to EVER append to an already existing PostSript file.
Perhaps you can explain the unusual sequence of events
that leads you to believe this is the case. As far as
I know, IDL *always* creates a new PostScript file.

Cheers,

David

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

Subject: Re: how to ensure overwriting of eps graphics files?
Posted by swingnut on Mon, 16 Apr 2007 12:15:33 GMT
View Forum Message <> Reply to Message

Here's a more thorough description of the behavior I observed.

My code builds a string that describes which settings have been set
when running the analysis. This string gets combined with a stamp
indicating what data file was used as input and a label indicating
type of plot in the file; this filename generation is automated and
repeatable. If the analysis is successful, subroutines are called to
do plotting. The point is to have an automated and completely
repeatable analysis so that anyone can run the code on the same data
and generate the same results I did, down to the names of the files
containing log output, calculated quantities, and plots.

I've been hitting the numerical experimentation hard to improve treatment of special cases and such. This does affect which data gets processed to the point of making plots, but I have certain subsets chosen so that if the code can't reach that point every single time, I have much bigger problems than my plots to fix. Thus there should be plots written to a known filename for every run of the code using one of those data sets. If I make a change to a plotting subroutine, I should see it the very next time I run the code since I have wrapper code that applies RESOLVE_ROUTINE to every .pro file in my code every single run. (This is the best way I've been able to come up with since I have to deal with IDLDE in windows. ::sigh:: Whole 'nother issue, don't want to go there.)

Anyways, I noticed this issue when I changed the way I was altering the calculation of the ranges for my plot axes this morning. The plot ranges were originally calculated using an operation that involved taking the max and min of the input data, taking the common logs of those, and rounding the values before storing in the variables used to set the xrange and yrange values. Originally these had the form

```
maxYcoord=10.0^CEIL(ALOG10(MAX(  )))
minYcoord=10.0^FLOOR(ALOG10(MIN(  )))
minYcoord=MAX([minYcoord,3.0])
```

with arguments supplied based on the quantities being plotted. These lines got changed to

```
maxYcoord=10.0^(1.05*ALOG10(MAX(  )))
minYcoord=10.0^(ALOG10(MIN(  ))/1.05)
```

because in general the range generated using the first approach spanned two decades or more, while all of my data lies within a single decade. While prototyping this change in only one plot routine to see if I liked the behavior, I noticed that the range of the y-axis wasn't always changing, despite my journaled log output telling me that the analysis had processed that set of inputs and that new plots should have been placed into those files. At this point, if I delete the files and rerun the analysis, then I get plots with the y-axis range generated using the new approach. I'm sure I might have missed something, but I can't think of an explanation other than IDL not actually overwriting the contents of existing postscript files sometimes.

Inside each of the plotting subroutines are the following bits of code, in case it helps to see what I'm doing:

```
;; Open the output file in the correct directory
```

```
CD,plotDir,CURRENT=oldDir
 DEVICE,/ENCAPSULATED,FILENAME=plotFile,/COLOR,BITS_PER_PIXEL =8,$
   PREVIEW=2,XSIZE=16,YSIZE=16

maxYcoord=10.0^(1.05*ALOG10(MAX(nDataLoc)))
minYcoord=10.0^(ALOG10(MIN(nDataLoc))/1.05)

;; Set up the plot axes and coordinates
plot,locRealHeights,nDataLoc,$
   /NODATA,$
   charsize=plotParamStruct.plotCharSize,$
   charthick=plotParamStruct.plotCharThick,$
   psym=4,$
   symsize=plotParamStruct.plotSymSize,$
   thick=plotParamStruct.plotSymThick,$
   background=colorStruct.white,$
   color=colorStruct.black,$
   yrange=[minYcoord,maxYcoord],$
    xrange=[minXcoord,maxXcoord],xtickformat='(f3.1)',XStyle=8,$
   xtitle='Distance (RE)',$
   ytitle='Densities (cm^-3)',$
   title='Field-aligned dens'

[rest of code in this file skipped, just overplots several stamps and
more data]

 ;;------------------------------------------------------------ ------------
;; Clean-up and exit
 ;;------------------------------------------------------------ ------------
DEVICE,/CLOSE_FILE
DEVICE,ENCAPSULATED=0
CD,oldDir
```

On Apr 15, 3:13 pm, David Fanning <n...@dfanning.com> wrote:
> swing...@gmail.com writes:
>> Ah, the never-ending challenges of working in IDL. I just discovered a
>> little while ago, while continuing to develop code for my
>> dissertation, that IDL will write a new postscript file if none
>> exists. Great. But in the process of debugging, setting reasonable
>> ranges for plotting, adding and removing items from various plots,
>> etc, I've stumbled upon an odd behavior whereby IDL is not overwriting
>> existing eps files with the updated plots. (I discovered this because
>> I changed something that altered which of the plots gets overwritten
>> based on the analysis.)

>
> I doubt this very, very much. I have never known IDL
> to EVER append to an already existing PostSript file.
> Perhaps you can explain the unusual sequence of events
> that leads you to believe this is the case. As far as
> I know, IDL *always* creates a new PostScript file.
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
