
Subject: IDL plot is inconsistent, where is subtle error?

Posted by [bleau](#) on Wed, 02 May 2001 17:20:02 GMT

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

Hello, IDLers. I'm having a very subtle problem with plotting from IDL. Either that or IDL is screwed up, which I doubt is the case. First, my configuration:

IDL Version 5.2 (vms alpha)

OpenVMS AXP V7.1-2

I'm plotting a set of values along the Y-axis (log plot), with time being on the X-axis. The time is expressed as a double precision floating point quantity (created with DBLARR). The time has a base year (time zero) of several decades ago, and I can't change this base year, so that's something I have to live with. This is called an epoch time. Typical values of an epoch time are in the range 357588000.D0. I have a routine that converts from epoch time value to normal calendar notation (date and time).

The length of the X-axis is 4 hours. I am using 4 major intervals (one hour each), for a total of 5 major tick marks, and 6 minor intervals. I am explicitly specifying to the PLOT command the values XTICKS (which is 5), XMINOR (which is 6), and XTICKV, which is a double floating vector of length 5. The vector I pass in for XTICKV contains five epoch time values, including the values at the two endpoints of the X-axis.

I also specify the function name 'XTICK3' (which I wrote) for the PLOT keyword XTICKFORMAT. XTICK3 is called by POT once for each time the X-axis has a major tick mark. XTICK3 labels all major tick marks. XTICK3 is passed the axis designation (X-axis is 0), tick mark index (range is 0 to 4), and value of that tick mark. The value, in this case, is the epoch time. XTICK3 decomposes the epoch time into hour, minute, second, creates a string of four characters containing the hour and minute (HHMM notation), and returns that string as the label to use at that tick mark. For a plot starting at 17:00 GMT, XTICK3 should return the values '1700', '1800', '1900', '2000', and '2100'. Note that this is effectively truncating the time to the preceding (next lower) minute.

My first version of XTICK3 did not act in this manner. I found, by experimentation, that the value passed to XTICK3 was a *single* precision floating value, even through all the X-values to be plotted, as well as the vector passed to XTICKV, were *double* precision. With the magnitude of the numbers I'm using for the epoch time, this resulted in a quantization effect, where the single precision epoch time was sometimes either greater or less than its corresponding double precision value by 16 seconds. This resulted in tick labels such as '1700', '1759', '1900', '1959', and so on.

I corrected for this by having XTICK3 add 30 seconds to the value passed to

it before converting it to hour, minutes, and seconds. This fixed the time axis labelling problem, while not changing the X values to be plotted, nor changing the X values where the major tickmarks are to be drawn.

Now to the meat of the reason for my post.

My current problem arises in that some of the time the last (5th) major tick mark does not get drawn or labelled. Through experimentation it seems that when the ending time is on an even hour only 4 major ticks are drawn and labelled, while when the ending time is on an odd hour the 5th major tick is both present and labelled. I have the tick marks drawn outward from the graph, so it is quite easy to tell when it is present or not.

IDL should not act in this manner. It should be consistent.

If you'd like to take a look at two sample plots here are URLs:

<http://umtof.umd.edu/sem/tickpresent.gif> <- works, odd hour
<http://umtof.umd.edu/sem/tickabsent.gif> <- fails, even hour

Here are some diagnostic dumps of data. First the case where it is working correctly:

```
range of x =      14400.000;      4.0000000 hours
X range is less than a day
xmin = 2001/05/01 15:00:00
xmax = 2001/05/01 19:00:00
n_intervals=      6      4
running computeticks
xmin= 2001/05/01 15:00:00
interval=      3600.00      1.00000 hrs
step=      0
  0 2001/05/01 15:00:00
  1 2001/05/01 16:00:00
  2 2001/05/01 17:00:00
  3 2001/05/01 18:00:00
  4 2001/05/01 19:00:00
done computeticks
```

```
Using      5 major ticks; values:
  0 2001/05/01 15:00:00      <- these are double precision epoch
  1 2001/05/01 16:00:00      times converted to Gregorian
  2 2001/05/01 17:00:00      notation
  3 2001/05/01 18:00:00
  4 2001/05/01 19:00:00
```

The epoch times shown below are the single precision values that correspond.

Idx	Epoch time	Mon	Dy	Hr	Min	Sec	Msec	Label
0 1	357577216.000	5	1	15	0	48	0	label='1500!C May 1!C doy121'
1 1	357580800.000	5	1	16	0	32	0	label='1600'
2 1	357584384.000	5	1	17	0	16	0	label='1700'
3 1	357588000.000	5	1	18	0	32	0	label='1800'
4 1	357591616.000	5	1	19	0	48	0	label='1900'

The vector dumped below is what is returned by the PLOT command using the keyword XTICK_GET.

```
tv=
  357577216.00
  357580800.00
  357584384.00
  357588000.00
  357591616.00
```

Here are some diagnostic dumps of data for the case where it is **not** working correctly:

```
range of x =      14400.000;      4.0000000 hours
X range is less than a day
xmin = 2001/05/01 16:00:00
xmax = 2001/05/01 20:00:00
n_intervals=      6      4
running computeticks
xmin= 2001/05/01 16:00:00
interval=   3600.00   1.00000 hrs
step=      0
  0 2001/05/01 16:00:00
  1 2001/05/01 17:00:00
  2 2001/05/01 18:00:00
  3 2001/05/01 19:00:00
  4 2001/05/01 20:00:00
done computeticks
```

```
Using      5 major ticks; values:
  0 2001/05/01 16:00:00      <- these are double precision epoch
  1 2001/05/01 17:00:00      times converted to Gregorian
  2 2001/05/01 18:00:00      notation
  3 2001/05/01 19:00:00
  4 2001/05/01 20:00:00
```

The epoch times shown below are the single precision values that correspond.

Idx	Epoch time	Mon	Dy	Hr	Min	Sec	Msec	Label
0 1	357580800.000	5	1	16	0	32	0	label='1600!C May 1!C doy121'

```

1 1 357584384.000 5 1 17 0 16 0, label='1700'
2 1 357588000.000 5 1 18 0 32 0, label='1800'
3 1 357591616.000 5 1 19 0 48 0, label='1900'
4 1 357595200.000 5 1 20 0 32 0, label='2000'

```

The vector dumped below is what is returned by the PLOT command using the keyword XTICK_GET.

```

tv=
357580800.00
357584384.00
357588000.00
357591616.00
357595200.00

```

Here is the PLOT command used that generates just the outline and the tickmarks (note the use of /NODATA). I use a subsequent PLOT command to place the data on the graph, and an AXIS command to draw the grid. (Please spare me suggestions about needing only one PLOT command, I really do need two, and that's not the issue here.) This command is in a loop with i=0,2 to be used to generate several graphs, hence some of its quantities being subscripted by i.

```

!x.minor = n_intervals(0)
!x.ticks = n_intervals(1)
!x.tickv = xtickvalues          ; xtickvalues is DOUBLE

plot, epoch, values(*,k), /nodata, $ ; epoch is vector of times
max_value = 9999998., $
xtype = 0, ytype = loglin(i), $
xrange = x_range, $           ; x_range is vector 2 long, see dump
yrange = y_range(*,i), $
title = graph_title + '' + namevec_title(i), $
xtitle = time_axis_title, $
ytitle = namevec_y(i), $
xcharsize = 0.75, $
xtickformat = x_label_routine, $ ; this has the value 'XTICK3'
xtick_get = tv, $              ; tv is returned as FLOAT vector
xticklen = -0.01, $
yticklen = -0.01, $
background = 255, color = 0

```

Lawrence Bleau
University of Maryland
Physics Dept., Space Physics Group
301-405-6223
bleau@umtof.umd.edu

Subject: Re: IDL plot is inconsistent, where is subtle error?
Posted by [Craig Markwardt](#) on Fri, 04 May 2001 18:50:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

bleau@umtof.umd.edu (Lawrence Bleau) writes:

> In article <gurman-669C26.14174604052001@news.gsfc.nasa.gov>, "Joseph B. Gurman"
<gurman@gsfc.nasa.gov> writes:

>>

>> But of course, Larry, you know that you can't create GIFs in 5.4....

>

> No, I did not realize this, and I just paid for a new copy (only \$10).

>

> Why did this happen? Is there a replacement operation? Is there a 3rd-party

> add-on to allow this?

Apparently the owner of the LZW compression "patent" -- Unisys --
popped a huge license fee increase on RSI, and RSI refused to pay.
Hence, no GIFs. This is despite the fact that the patent is only
valid for another year or so.

There has been some success with people transferring the GIF DLM's
from an older version of IDL to the newer version. Sometimes it
works, sometimes not.

Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
