Subject: Re: Top 10 IDL Requests
Posted by Mark Fardal on Thu, 03 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Hi,

I am sort of out of the loop, chugging along with limited knowledge
of the features in 5.2.1, but here are my wishes:

1. Update structures without quitting idl, and without resetting
entire session either. Same for common blocks. David F. keeps
casting aspersions on these ideas; but if this behavior were changed
maybe some people would actually program with objects... Besides, the
whole point of objects is that data and methods are supposed to be on
equal terms. So why can you add a new method, but not a new data
element, to an object?

2. arrays: Craig had particularly good suggestions here. In fact I want
to say "me too" to his whole post.
allow zero-length arrays
STRICT keyword to constructors
don't alter dimensionality on type conversion. You don't alter the
type when you change the dimensionality, after all.

3. Better plots; fix long-standing bugs and improve default behavior.
There are a bunch of small things here:

[double precision--oh wait, it's done!]

Limit on number of zeros used for floating point tick labels. I haven't
figured out IDL's default algorithm but it needs some work. E.g.
plot_oo,10.^findgen(10),10.^(0.6*findgen(10)-4), ytitle='Invisible'

Fix bug creating a log axis. Try
plot_oo,[.1,1],[.1,1],ystyle=1+8
axis,/yaxis,yrange=[3,4],ylog=0, ytitle='should be linear
You get a log axis from 1 to 10 on the right, though you asked for linear.

Fix bug in behavior of multiline titles on upper X axis. To see
what I mean, type
plot, findgen(100), position=[0.2, 0.2, 0.8, 0.8], /norm, $
xstyle=8, ystyle=8, $
xtitle='X axis title: one!Ctwo!Cthree', $
ytitle='Y axis title: one!Ctwo!Cthree'
axis, /xaxis, xrange=[0,1], xtitle='X axis title: one!Ctwo!Cthree'

axis, /yaxis, yrange=[0,1], ytitle='Y axis title: one!Ctwo!Cthree'

Log plots with zero or negative values: use lowest positive value,
not 1.e-12, as lower limit.

Label minor ticks on log plots when necessary.
plot_oo,10.^findgen(10),10.^(0.05*findgen(10)+0.5),yr=[2,9], /yst
Yes, I know, this is a silly range. But even so, you should still
have the information needed to read the plot.

Independent plotting system variables for each graphics window. Not
sure how this should be implemented but it would definitely be useful.

4. New operators
Separate boolean and bitwise operators.
Bitwise "and, or, xor, not" could be & | ^, (! or ~) respectively
Boolean operators would just be and, or, xor, not
then redefine "true" to be any non-zero value! (I'm dreaming...)
C-like arithmetic operators: +=, ++, etc

5. HISTOGRAM routine:
keyword to use flexible (variable-spacing) bin boundaries.
keyword to add empty bins to ends (useful for plotting)
keyword to return bin centers

6. A fast routine to read a columnar text file, as a standard part of
the IDL distribution. Or am I missing one that now exists? I know
there are a lot of publicly available routines, but this should come
with IDL--it's about the first thing most users want to do. Besides,
even the fastest routines I know of still run many times slower than
SM does.

7. Don't quit out when there is a Ctrl-D at beginning of line. I have
hit this by accident many times, especially in emacs/IDLWAVE, and it's
always a pain. If this offends some people, perhaps offer them a
choice--e.g. through an environment variable IDL_IGNOREEOF on unix (I
suppose other platforms have similar concepts).

8. Improve accuracy, stability, user interface, and documentation of
math routines. I don't have specific complaints at the moment,
as the problems I've run into in the past may well be fixed now.
But the history of the math routines in IDL is not good. This
is reason #2 I can't recommend IDL to other people.

9. An actual RSI presence on this newsgroup. Preferably, have a designated point man / flak-catcher. I have seen newsgroups where this strategy was adopted; for example, Jens Alfke at the Mac Java mailing list and Ron Liechty at the Metrowerks newgroup fulfilled this role, once upon a time. It made users very happy.

10. Lower prices, particularly on the multiple site licenses. This is reason #1 I can't recommend IDL to other people; they all think it's too expensive. Which means I can't share code with other people. Which means I have less of an incentive to write things in IDL. Which means I have more of an incentive to move to something else. I think that with lower prices, there could be a phase transition in the number of users, so it doesn't necessarily mean lower revenue for RSI.

Mark Fardal
UMass
