Subject: Top 10 for old farts
Posted by Joseph B. Gurman on Fri, 28 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.13e947d0f2a8c1bf989b70@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:

> Joseph B. Gurman (gurman@gsfc.nasa.gov) writes:
>
> > And I'm very serious about the point above. I may be stuck knowing
> > a
> > bunch of old farts (not, actually), but maybe one person in twenty
> here
> > actually uses the object capabilities when given a choice.
>
> Oh, I don't doubt your figures at all. I just think it
> is a shame, given how easy objects are to use (and I am
> *not* talking about object graphics now) and how
> significantly they could change the way your write
> programs.
>
> But I'm old enough to remember how we had to bring
> all you old farts, uh, excuse me, scientific programmers,
> along with widgets, too, so I still hold out hope. :-)
>
> Cheers,
>
> David

David -

I guess you've seen the responses from Mark Hadfield and Luis Alonso on the overhead involved in using objects. A colleague at another instituion e-mailed me to say:

> I think your
> views represent those of many of us older folks trying to use
> IDL. I've messed with objects in IDL enough to realize they will
> probably destroy what little productivity I seem to have left as
> a scientist. Just give me the data arrays--hopefully processed
> into a scientifically useful form--and let me get on with the
> science part.

(I believe this "older folk" is about 49, or 137 in programmer years.)

The real beauty of IDL for scientists is _not_ its ability to do everything in the most elegant way possible, but in its capability to do 90% of what we want _very fast_ and to do more elegant things on a

time-invested basis (e.g. really pretty plots; 3D shaded, rotating surfaces; the ability to save such projections as objects).

For most scientists, at least astronomers, the three things they spend most of their time on are:

1. writing proposals

2. writing proposals

3. writing proposals

Doing research and writing appears sadly comes in fourth; dealing with silly bureaucracies fifth; and exploring the programming capabilities of IDL, perl, tcl, ... somewhere in the nth category.

And it's the grants we get with our proposals (occasionally) that pay for the IDL licenses.... and maintenance.... and training.... and even ;-) books.

The grants also pay for graduate students' and programmers' time, which can be leisurely (and no doubt worthwhile-ly) spent on object programming in IDL.

So I still propose that as long as there are lower-priced, full-featured student licenses, there should be lower-priced, fewer-featured research associates' licenses. The "pro" license can include all the wonderful features those with time to use them efficiently want.

Seriously (once again), it would be nice to be able to pay for a base license, and add on, at extra cost:

1. objects

2. QuickTime support (per codec)

3. other features requiring RSI to pay license fees (GIF?)

BTW, the two people here (one Ph.D., one graduate student) who do use objects, and use them well, are both on vacation now, so they're not around to contradict me (heh, heh).

Also BTW, I asked only for _flames_ to be directed to /dev/null, as I recall.... and David never flames anyone.

Best,

Joe

--
| Joseph B. Gurman, NASA Goddard Space Flight Center, Solar Physics
| Branch, Greenbelt MD 20771 USA / Federal employees are still
| prohibited from holding opinions while at work. Therefore, any
| opinions expressed herein are somebody else's.

Subject: Re: Top 10 for old farts
Posted by Pete Riley on Sat, 29 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Joseph B. Gurman in a flash of brilliance said:

> The real beauty of IDL for scientists is _not_ its ability to do
> everything in the most elegant way possible, but in its capability to do
> 90% of what we want _very fast_ and to do more elegant things on a
> time-invested basis (e.g. really pretty plots; 3D shaded, rotating
> surfaces; the ability to save such projections as objects).

Joe:
I agree 101%. But, I think the larger picture here though is that RSI missed the "object" boat. When "Tek" terminals were in vogue, direct graphics were perfectly adequate. But since the early 90's, graphics packages that *really* use object concepts have performed basic graphing tasks much better than anything in IDL. I'm constantly amazed by the number of Mac users (scientists) still propagating their copy of Kaleidagraph to their next powermac. I can't blame them though; Insight doesn't come close to mimicking the capabilities of a > 10 year old program. RSI is struggling with the object paradigm shift. I believe it was a mistake to introduce the object graphics so early and, at the same time, maintain the direct graphics. Why not replace all the direct graphics calls with object equivalents? You should be able to say:

PLOT, findgen(10)

and you get an object window (maybe plus the print, modify,etc...options that are in insight). There should have been a transparent migration to object graphics. I can appreciate that there are technical difficulties. But as an end user, they shouldn't affect ME.

-Pete Riley

Subject: Re: Top 10 for old farts
Posted by davidf on Sun, 30 Jul 2000 07:00:00 GMT

Mark Hadfield (m.hadfield@niwa.cri.nz) writes:

> Objects were a necessary development in IDL and are certainly a good thing
> IMHO. Well, OK some of the design decisions were debatable.
>
> Object graphics were also a necessary development but are less obviously a
> good thing. The main problem with them is that producing a simple plot using
> IDL's standard object graphics facilities is ridiculously difficult. It's
> possible to finish RSI's job by writing a set of smarter, higher-level
> graphics classes & routines but it's a lot of work. Hence my comment about
> productivity.

Amen to this. But I would like to reiterate (for what, the
fourth time? I've got to give up on this horse) that they
are not so difficult that *someone* (not RSI, apparently)
wouldn't be willing to go to the trouble if there was
some indication they would be compensated for the effort.
And I mean by "compensated", rewarded in some other way
than by the eternal gratitude of the IDL newsgroup
community, with all the resultant adulation and prestige
(which doesn't count for much with the bank manager) such
acknowledgement implies. :-)

Cheers,

David

P.S. I've already got Joe's order written down. Anyone
else ready to anti-up for some decent (direct or object)
graphics programs, written as objects? In the meantime
I'm going to get on with my life. You will hear no more from
me on this subject. (Unless someone provokes me, of course.)

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Top 10 for old farts
Posted by davidf on Sun, 30 Jul 2000 07:00:00 GMT

Since I enjoy (as Brian Jackel puts it) a "pointless, but

interesting discussion" almost as much as I do writing IDL object programs, let me just weigh in with one more response.

Pete Riley (uk2@linuxfan.com) writes:

> Why not replace all the direct graphics calls with object
> equivalents? You should be able to say:
>
> PLOT, findgen(10)
>
> and you get an object window (maybe plus the print, modify,etc...options
> that are in insight). There should have been a transparent migration to
> object graphics.

This was exactly RSI's original intention. But, alas, things don't always go the way we plan. And with object graphics the vagaries of programming projects were even more pronounced than usual. It was almost as if Coyote himself were put in charge of it.

Let me recount the story as I remember it. (Which is to say, this may not be how it *really* happened. But it is the way I tell the story.)

Some four or five years ago now RSI realized that their graphic system was out of date. It was wonderful for the early 1980's, but these were the 90's and the millennium was fast approaching. And in particular, their 2.5D way of rendering surfaces (the Z axis must always be vertical) was hopelessly out of date. Three orthogonal axes should be completely rotatable. The decision was made to completely overhaul the entire graphic system and bring it up to date.

One of the best 3D graphic guys in the country (I forget his name) was hired to design the system and the RSI programmers set to work. A year was set aside for the effort, during which time there would be no work done on bug fixes etc., as everyone was tied up in building this new thing. Customers would get jumpy about paying maintenance dollars during this time, but the new system would be so much better than the old that this problem was judged worth the risk.

To make a long and painful story short, it took a LOT longer than one year to do the work. In fact, it was nearly two years by the time IDL 5.0 left the barn. Customers were screaming, maintenance revenue was dwindling, new license sales were off as customers

were waiting for the big new release that was right around the corner. And expenses were up sharply as more programmers were hired to keep up with the larger and larger work load. I think everyone in the building was putting in 60+ hour weeks. In short, it was one of those really tense, trying times that all businesses go though occasionally.

But about 6 months before the actual release the graphics system was built sufficiently (most of the effort up to then had gone into coding the graphics primitives that could be assembled into line plots, surface plots, etc.) that simple "high level" graphics commands could be tested. The idea had always been (I think because this is what the graphics guy told us) that 2D graphics would "fall out of" the 3D graphic system. In other words, 2D graphics were just a specific subset of the 3D system.

But that turned out to be a bit of an oversimplification. Because when the first line plot was tested in the new graphics system it rendered about 10 times slower than the equivalent direct graphics command. In fact, *all* of the 2D commands were about 10 times slower than the equivalent direct graphics commands. The problem, as it turned out, is inherent in 3D systems, and cannot be easily solved even today, I think. It is the reason you have never been able to make a decent line plot in a fabulous 3D graphics system like AVS.

Oh, oh. We were going to go to angry customers and tell them "OK, you paid us a lot of maintenance money for two years and didn't receive anything for it, but here is your great new system. And by the way, it is 10 times slower than the old one." It was going to be a hard sell, to say the least. :-)

So the decision was made (the correct one, I think) to keep the old direct graphics system, but to add the new object graphics system to IDL. Customers could use the one that made the most sense to them. It was expected that most line plots and image displays would be done in the old direct graphics system (since it was a lot faster than the object graphics system), but that visualizations that required a robust 3D capability would use the object graphics system.

That is pretty much where we are today.

Where RSI has fallen short, I think, is in not making more high-level object graphics commands. It is still pretty much a system that has to be built from scratch. (I know there is IDL Insight. But have you used it!? I find it infinitely frustrating and completely non-intuitive. Plus, aesthetically it is awful. I'm embarrassed by it whenever I have to show it to someone.) And no one else has stepped forward really to make the kinds of tools that people need. (Partly, I should tell you, because no one seems to want to pay you for making the effort. :-)

My sense of what is happening in object graphics is that RSI is moving on to other things (I don't see much in the way of updates on the list of features for IDL 5.4.) Perhaps they are locked into Insight for political reasons, I don't know. Or perhaps there are more people using it than I realize. (Lord knows I couldn't abide a Top 10 Things I Like About Insight thread.)

But, in any case, objects (which had to be invented to create the object graphics class library) are NOT object graphics. Too many people fail to make the distinction. When I talk about object programming I am very rarely talking about using the object graphics class library. I'm taking about using a new way of constructing programs that has the potential to make your programs much more useful to you.

This ability to create your own objects is the major story, I think, in the IDL 5.0 release. Not the object graphics systems. I don't know why it has fallen on me to be the spokesman for the Crusade to Get the Object Story Out, but it appears to be the case. But then Coyote has always liked his little jokes. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Top 10 for old farts
Posted by Joseph B. Gurman on Sun, 30 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.13eb4d1b4b64e350989b77@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:

[snip]

> In fact, I haven't written a program for a client in the
> past year that hasn't included at least one object,
> and sometimes it's easier to write the whole thing as
> an object. PSCONFIG, a program on my web page that is
> widely downloaded and praised by people who use it, is
> an object program, although I would guess 95% of the
> people who use it don't know that. Certainly most of
> the programs I add to my library in the future will
> be object programs, although I'm not fool enough to
> tell anyone that, since just the word itself is enough
> to send shivers up the backs of many IDL users. :-)

I believe, I believe, I believe. I just have no need (yet) to write object code myself.

>
> I've frankly pretty much given up the idea of writing
> an object book because (1) it is so damn hard to write
> a book, and (2) after going to all that trouble I thought
> only about a dozen people would buy it. (And I will hear
> from all 12 today, probably, pleading with me to reconsider,
> so desperate is the need for decent documentation.)

Well, you might sell a 13th, just because you've piqued my interest, but I would put it in the "programming to learn" pile, right under the 10%-finished Perl book, and the 0% finished Palm OS book. Frankly, I think I'd rather learn to write Palm OS apps (so I can analyze data anywhere), but that's another story.

>
> The lack of good instruction is probably what is holding
> the adoption of objects back. I know you say you don't
> need them. The people in my course last week were adamant
> that they didn't need to know any widget programming, too.
> But in the end all they cared about was widget programming
> and increased interactivity with their data. I submit that
> objects have the same ability to transform how we work
> with our data.

Careful what you say, Dave, because the truth can hurt. I _use_

widget apps every day --- that I specified to a good widget programmer years ago, and that get maintained sporadically. If you or your book can convince me I need some capabilities of IDL objects I haven't seen yet, I might ask a widget programmer to gin something up for me.... but it would have to be an awfully big something. You see, I went to the trouble (during a six-month sabbatical in France 8 years ago) to learn the original widget API.... and then they went and changed it. (In the meantime, I'd written some pretty neat widget code, IMHO.) In the words of Boris Badanov, "Fooey and double fooey." Just not enough incremental functionality to make me learn the whole, new interface when we had realtime data acquisition and reformatting software (the latter in IDL, no widgets) to write, prototype, and implement, and a satellite to launch.

>
> Yes, there is more up-front cost. And I'm completely
> sympathetic with those scientists who feel they don't
> have time to figure it out. If objects are unfamiliar
> to you, it is just about impossible to learn about them
> from the material that RSI gives you. But in the end
> they do more for your science than what you are using
> now. That must be the bottom line. Someday, inevitably,
> you are going to be working with objects. Maybe you
> won't write them. But you sure as heck don't want to
> pay extra for them, either.

Yes, I do want to pay "extra" for them, if for the same delta I can get QuickTime movies and interlaced GIFs instead. Honestly.

As for the argument that RSI shouldn't charge on a by-not-so-useful-to-everyone feature basis, I view IDL rather like a power tool. If Black and Decker or Makita wants to add lots of bells and whistles to their new model of electric screwdriver, that's great, but I'd rather they simply make them all options that I can buy if I want or need them --- and by and large, they do. Now I know some people worship their power tools, but I find them fun and useful --- just like the parts of IDL I use. Well, maybe "fun" is too strong a word for some IDL "features," but....

Since I leave for the IAU meeting in the UK and then vacation (and hopefully Intenret abstinence) for the next three weeks starting Wednesday, I won't dribble any more gasoline on this particular fire. The RSI lurkers can gauge how many people feel this way or that.

Best,

Joe

P.S. My favorite IDL 5.3 "feature" (tested only in OpenVMS 7.2-1 so
far): including the statement

device, cursor_standard = 68

in your IDL startup file reduces your available pixmap memory (e.g. for
XINTERANIMATE movies) by 50%. What a hoot! Those programmers at RSI have
such a sense of humor. (No such ill effect is seen in earlier versions
of IDL.)

Subject: Re: Top 10 for old farts
Posted by Joseph B. Gurman on Sun, 30 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

In article <3981E962.9AA8D07B@cmdl.noaa.gov>, promashkin@cmdl.noaa.gov
wrote:

> "Joseph B. Gurman" wrote:
>> For most scientists, at least astronomers, the three things they
>> spend most of their time on are:
>>
>> 1. writing proposals
>>
>> 2. writing proposals
>>
>> 3. writing proposals
>
> If so, then a good word processor is all that is actually needed :-)
>
> Cheers,
> Pavel

Got that, too, and it's a lot cheaper than IDL. On a dollar return
basis, I guess it's better software. ;-)

Fortunately, there are other discriminators than money.

Joe

Subject: Re: Top 10 for old farts
Posted by davidf on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Paul van Delst (pvandelst@ncep.noaa.gov) writes:

> P.S. Who's this Coyote?

Coyote is well known to astronomers. When the Old People
were carefully placing the stars into the sky in these
beautiful constellations, Coyote stumbled into the teepee
from a night out carousing and knocked the sack of stars
everywhere. That's why it's such a jumble up there now.

Cheers,

David

P.S. Let's just say Coyote is inordinately fond of
hanging out with IDL programmers. :-(

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Top 10 for old farts
Posted by davidf on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Michael Cugley (mjcugley@tigger.medschool.dundee.ac.uk) writes:

> As a newbie to IDL, but with some graphics and OOP experience, what
> kind of programs are needed?

Basically, any kind of graphics program that can draw itself
both to the display and to a PostScript file. A bit of
interactivity is nice (update axis range, change the color
of symbols, etc.), but is not as important as getting the
darn thing into a PostScript file in more or less the same
way it is displayed on the screen.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Top 10 for old farts
Posted by Paul van Delst on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Pavel Romashkin wrote:
>
> Paul van Delst wrote:
>
>> Ahhhhh. When IDL v5 was released, the first thing I did was convert some
>> of my code to use OG. My god was it slow. Also, when I started up
>> Insight I figured that it had to be the showcase of what IDL OG had to
>> offer. That all pretty much turned me off OG - and thus IDL objects - in
>> general. The latter was a bad decision I know, but my "to-do" list kept
>> getting bigger during the 2-week period I spent playing with IDL OG.
>
> I am not arguing about IDLgr* speed with Paul, or, God forbid, David,
> but I find that with today's computers even object graphics is not that
> slow. I don't use multitudes of graphic objects, but rendering a few
> line plots with zoom, point selection, deletion, calculation, adding and
> removing temporary grPlots, etc. - I notice so little difference
> compared with direct graphics that it is not detrimental to the
> application. I think that once faster machines become more widespread,
> 3D object graphics will find more use. On a Pentium 60 w/o OpenGL, I
> agree, it is intolerable.

Maybe for 99% of cases you're right and I fall in the remaining 1%? :o)

The test I applied was to plot what I usually plotted day-to-day with DG and OG. What I usually plot are high resolution atmospheric radiance and/or transmittance spectra - sometimes 100,000's points per spectrum, with, oh, about 40-100 spectra per atmospheric profile. Plotting this with DG took a long time (seconds) but plotting in OG - well I could go and answer the call of nature, return and it would still be in the process. I'm sure part of it is due to the way I put the plotting code together - remember this was my learning experience - but after a couple of days I started thinking that one of the main reasons I bought IDL was so I could type:

PLOT, x, y

and, shazaam! there was my data on screen. With all the programming required to get a plot with OG - onscreen or on paper - I may as well buy a F90 graphics library for my system (non-portable, granted) and do the number crunching AND the plotting in F90. (I haven't had to write code to scale data axes for plotting since I linked up an old tektronix

plotter to a VAX terminal back in old day...well, mid-80's)

In this situation, the destination was more important than the journey.

paulv

p.s. The system was a Sun dual 250MHz processor with the optional
Creator 3-D graphics accelerator and 512MB of RAM. I don't know if the
graphics extras helped the IDL OG plotting speed or not, and although
it's not a super flash machine, it was pretty zippy. So, given all that,
I couldn't justify using IDL's OG at all.

Maybe things have changed since IDL v5.0 but the first impression has
stuck like sh*t to a blanket. :o)
--
Paul van Delst Ph: (301) 763-8000 x7274
CIMSS @ NOAA/NCEP Fax: (301) 763-8545
Rm.202, 5200 Auth Rd. Email: pvandelst@ncep.noaa.gov
Camp Springs MD 20746

Subject: Re: Top 10 for old farts
Posted by davidf on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Pavel Romashkin (promashkin@cmdl.noaa.gov) writes:

> I am not arguing about IDLgr* speed with Paul, or, God forbid, David,
> but I find that with today's computers even object graphics is not that
> slow. I don't use multitudes of graphic objects, but rendering a few
> line plots with zoom, point selection, deletion, calculation, adding and
> removing temporary grPlots, etc. - I notice so little difference
> compared with direct graphics that it is not detrimental to the
> application. I think that once faster machines become more widespread,
> 3D object graphics will find more use. On a Pentium 60 w/o OpenGL, I
> agree, it is intolerable.

Surely even NOAA can afford to buy Paul a better machine
than that! :-)

My personal opinion is probably close to Pavel's. I don't
think speed has held up the acceptance of object graphics.
The speed is surely acceptable. It has been the hours it
takes to put together a simple plot from the ground up,
and then the lack of printing capability once you got there.

Vector printing capability has fixed part of the problem,
but we are still faced with too few higher-level programs.

Cheers,

David

P.S. I suppose a case can be made that having Insight
as the flagship example of object graphics programming
has set the cause back 5-10 years, but I'm going to
stay out of it. :-)

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Top 10 for old farts
Posted by promashkin on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Paul van Delst wrote:
> Throwing
> money and people at a problem when things get tight - sheesh. Is this
> still an accepted mode of operation at software houses

I wish I could join Paul in sighing about inacceptability of this
policy, but here on Earth this is how it is done (if the result is
needed), not only in software companies. Well, not at NOAA of course,
but by other establishments that have money :-(

> Ahhhhh. When IDL v5 was released, the first thing I did was convert some
> of my code to use OG. My god was it slow. Also, when I started up
> Insight I figured that it had to be the showcase of what IDL OG had to
> offer. That all pretty much turned me off OG - and thus IDL objects - in
> general. The latter was a bad decision I know, but my "to-do" list kept
> getting bigger during the 2-week period I spent playing with IDL OG.

I am not arguing about IDLgr* speed with Paul, or, God forbid, David,
but I find that with today's computers even object graphics is not that
slow. I don't use multitudes of graphic objects, but rendering a few
line plots with zoom, point selection, deletion, calculation, adding and
removing temporary grPlots, etc. - I notice so little difference
compared with direct graphics that it is not detrimental to the
application. I think that once faster machines become more widespread,
3D object graphics will find more use. On a Pentium 60 w/o OpenGL, I
agree, it is intolerable.

Cheers,
Pavel

Subject: Re: Top 10 for old farts
Posted by Paul van Delst on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Pete Riley (uk2@linuxfan.com) writes:
>
>> Why not replace all the direct graphics calls with object
>> equivalents? You should be able to say:
>>
>> PLOT, findgen(10)
>>
>> and you get an object window (maybe plus the print, modify,etc...options
>> that are in insight). There should have been a transparent migration to
>> object graphics.
>
> This was exactly RSI's original intention. But, alas, things
> don't always go the way we plan. And with object graphics the
> vagaries of programming projects were even more pronounced
> than usual. It was almost as if Coyote himself were put in
> charge of it.
>
> Let me recount the story as I remember it. (Which is to
> say, this may not be how it *really* happened. But it is
> the way I tell the story.)
>

[snip...]

> To make a long and painful story short, it took a LOT
> longer than one year to do the work. In fact, it was
> nearly two years by the time IDL 5.0 left the barn.
> Customers were screaming, maintenance revenue was
> dwindling, new license sales were off as customers
> were waiting for the big new release that was right
> around the corner. And expenses were up sharply as
> more programmers were hired to keep up with the
> larger and larger work load. I think everyone in the
> building was putting in 60+ hour weeks. In short, it
> was one of those really tense, trying times that
> all businesses go though occasionally.

This reads like the classic "bad planning from the start" case study one finds in books about software project management, i.e. how *not* to do it (read ch8 of Steve Maguire's book "Debugging the Development Process". The chapter is titled "That Sinking Feeling" :o). Throwing money and people at a problem when things get tight - sheesh. Is this still an accepted mode of operation at software houses (anyone see the PBS special on Netscape just before they went open source? My god, wot a zoo.) ?? I realise that there are other pressures on a business but seeing that this type of modus operandi is still kicking makes the rapid movement of people in the software programming/design/engineering industry more understandable.

All that said, I think IDL is still a top-quality product. I remember when I was first looking into getting some sort of data visualisation software, the two choices were IDL or Matlab. Matlab was *so* much cheaper than IDL until you realised you had to buy all these toolkits to achieve the same functionality. After that the only reason Matlab was still competitive ($$-wise) was because the uni I worked at had a site license.

> Oh, oh. We were going to go to angry customers and tell
> them "OK, you paid us a lot of maintenance money for
> two years and didn't receive anything for it, but here
> is your great new system. And by the way, it is 10 times
> slower than the old one." It was going to be a hard sell,
> to say the least. :-)

Ahhhhh. When IDL v5 was released, the first thing I did was convert some of my code to use OG. My god was it slow. Also, when I started up Insight I figured that it had to be the showcase of what IDL OG had to offer. That all pretty much turned me off OG - and thus IDL objects - in general. The latter was a bad decision I know, but my "to-do" list kept getting bigger during the 2-week period I spent playing with IDL OG. :o(

anyway...

paulv

P.S. Who's this Coyote? I guess I should follow the off topic links on DF's website and read about him again (it's been a while).

--
Paul van Delst Ph: (301) 763-8000 x7274
CIMSS @ NOAA/NCEP Fax: (301) 763-8545
Rm.202, 5200 Auth Rd. Email: pvandelst@ncep.noaa.gov
Camp Springs MD 20746

Subject: Re: Top 10 for old farts
Posted by Ben Tupper on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:

> ....let me just weigh in with one more response....
>

Good Morning,

That is infomative and thought provoking.

Thanks for shedding some light on this history.

Ben

--
Ben Tupper
Bigelow Laboratory for Ocean Science
West Boothbay Harbor, Maine
btupper@bigelow.org
note: email address new as of 25JULY2000

Subject: Re: Top 10 for old farts
Posted by Martin Schultz on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Joseph B. Gurman (gurman@gsfc.nasa.gov) writes:
>
>
>> Seriously (once again), it would be nice to be able to pay for a
>> base license, and add on, at extra cost:
>>
>> 1. objects
>>
>> 2. QuickTime support (per codec)
>>
>> 3. other features requiring RSI to pay license fees (GIF?)
>
> I suppose it is inevitable, as IDL grows ever larger, that
> we begin to pay for add-ons. RSI has already taken this
> course with DataMiner and the Wavelet Toolkit. But I am
> dead set against this proposal, Joe.
>

> I submit that
> objects have the same ability to transform how we work
> with our data.
>
>
>
> Cheers,
>
> David

I strongly agree with David here! Although I must admit that I have not yet developed one really functional object application (just too much "Need this yesterday!" stuff), the availability of objects surely changed my perception of data analysis and allows for a much better strategic planning of data analysis software - even if in the end it is written as a "classical" program. At least in my case, there is still a gap between my mind and my fingers: I am thinking objects now, but I am still typing a lot of non-object programs. Yet I hope that my fingers will catch up within the next year or so ...

I also would like to direct your attention to the evolvement of FORTRAN which in some ways can be percepted to be the most conservative programming language one can think of (tons of legacy code, no one really learns it these days, so people learn it from looking at - yes! - legacy code ;-) So far, all global atmospheric model sources that I have seen have their origin in FORTRAN77 code (or even earlier). But, I also see many groups reworking their code to FORTRAN90, and progressively using modules. This could really be seen as a big step towards object oriented programming, because modules are designed to provide some of the major object features: (1) reusability, (2) encapsulation, (3) grouping of things that belong together. Now, try to think 3 (or maybe 5) years ahead and let people analyze their model results. Would they want to use an IDL that does not support objects, that forces them to go back in time?

Cheers,
Martin

--
[[[[[[[[[
[[Dr. Martin Schultz   Max-Planck-Institut fuer Meteorologie   [[
[[Bundesstr. 55, 20146 Hamburg   [[
[[phone: +49 40 41173-308   [[
[[fax: +49 40 41173-298   [[

[[martin.schultz@dkrz.de [[
[[[[[[[[[

Subject: Re: Top 10 for old farts
Posted by Michael Cugley on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

davidf@dfanning.com (David Fanning) writes:

> P.S. I've already got Joe's order written down. Anyone
> else ready to anti-up for some decent (direct or object)
> graphics programs, written as objects?

As a newbie to IDL, but with some graphics and OOP experience, what kind of programs are needed?

--
Michael Cugley (mjcugley@medphys.dundee.ac.uk)

Subject: Re: Top 10 for old farts
Posted by Mark Hadfield on Mon, 31 Jul 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Joseph B. Gurman" <gurman@gsfc.nasa.gov> wrote in message
news:gurman-FF65F5.09515928072000@news.gsfc.nasa.gov...
> I guess you've seen the responses from Mark Hadfield and Luis Alonso
> on the overhead involved in using objects.

Speaking only for myself (not Luis) I should elaborate.

As David has already pointed out so well elsewhere in this thread

"OBJECTS" does not equal "OBJECT GRAPHICS"

Objects were a necessary development in IDL and are certainly a good thing IMHO. Well, OK some of the design decisions were debatable.

Object graphics were also a necessary development but are less obviously a good thing. The main problem with them is that producing a simple plot using IDL's standard object graphics facilities is ridiculously difficult. It's possible to finish RSI's job by writing a set of smarter, higher-level graphics classes & routines but it's a lot of work. Hence my comment about productivity.

Mark Hadfield
m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield/
National Institute for Water and Atmospheric Research
PO Box 14-901, Wellington, New Zealand

Subject: Re: Top 10 for old farts
Posted by Nigel Wade on Tue, 01 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Pete Riley wrote:
>
> Joseph B. Gurman in a flash of brilliance said:
>
>> The real beauty of IDL for scientists is _not_ its ability to do
>> everything in the most elegant way possible, but in its capability to do
>> 90% of what we want _very fast_ and to do more elegant things on a
>> time-invested basis (e.g. really pretty plots; 3D shaded, rotating
>> surfaces; the ability to save such projections as objects).
>
> Joe:
> I agree 101%. But, I think the larger picture here though is that RSI missed
> the "object" boat. When "Tek" terminals were in vogue, direct graphics were
> perfectly adequate. But since the early 90's, graphics packages that
> *really* use object concepts have performed basic graphing tasks much better
> than anything in IDL. I'm constantly amazed by the number of Mac users
> (scientists) still propagating their copy of Kaleidagraph to their next
> powermac. I can't blame them though; Insight doesn't come close to mimicking
> the capabilities of a > 10 year old program. RSI is struggling with the
> object paradigm shift. I believe it was a mistake to introduce the object
> graphics so early and, at the same time, maintain the direct graphics. Why
> not replace all the direct graphics calls with object equivalents? You
> should be able to say:
>
> PLOT, findgen(10)
>
> and you get an object window (maybe plus the print, modify,etc...options
> that are in insight). There should have been a transparent migration to
> object graphics. I can appreciate that there are technical difficulties. But
> as an end user, they shouldn't affect ME.
>
> -Pete Riley

If you want this functionality you could use MATLAB... ;)

But I think the important thing is that we need the /choice/.

I have some (limited) experience of this problem with MATLAB. I was involved in a software development project and one of the modules was a visualisation tool for scientific data. For various reasons MATLAB was the chosen platform.

Some way along with the development it became apparent that MATLAB was not going to be able to do what was required. The graphics required drawing 100,000's of filled rectangles, and because of the object nature of MATLAB graphics the memory requirements to store all the graphics objects became too large for any reasonable system. Now, MATLAB has no other graphics system, so we had no choice but to drop MATLAB in favour of IDL direct graphics.

If IDL had only had object graphics it, too, would probably have been unable to do what we required.

So, object graphics is fine for those who require it; but it's also necessary to keep direct graphics for others who have different requirements.

--

Nigel Wade, System Administrator, Space Plasma Physics Group,
University of Leicester, Leicester, LE1 7RH, UK
E-mail : nmw@ion.le.ac.uk
Phone : +44 (0)116 2523568, Fax : +44 (0)116 2523555

Subject: Re: Top 10 for old farts
Posted by davidf on Wed, 02 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Luis Alonso (luis.alonso@uv.es) writes:

> PD: is that the right expression 'good night and don't let the bed bugs bite
> you'? My english's expressions are not in shape... i keep on shuffling the
> words :\

Close enough. And not bad for closing in on midnight. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Top 10 for old farts
Posted by Luis Alonso on Wed, 02 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Mark Hadfield" <m.hadfield@niwa.cri.nz> wrote in message
news:964994988.21151@clam-ext...
> "Joseph B. Gurman" <gurman@gsfc.nasa.gov> wrote in message
> news:gurman-FF65F5.09515928072000@news.gsfc.nasa.gov...
>> I guess you've seen the responses from Mark Hadfield and Luis Alonso
>> on the overhead involved in using objects.
>
> Speaking only for myself (not Luis) I should elaborate.

Hi, now is my turn to express myself (after digesting the huge amount of submissions to the newsgroup).

I think after all this discussion it's pretty much clear that objects (and not only O.G.) are a must in IDL, in the same way that the most classical way of programing (included direct graphics).

As I said previosly, i've playing around with object graphics, but not been able to do much by myself, so i kept using D.G. for those 'i need them for yesterday' plots, but for interactivity O.G. were a must, you know the kind of 'spin me this 5000 facets 3D modell around while zooming in and out, and change the colours cause those don't look the way i like' stuff.

Today i've spent a whole afternoon (from 3pm to 21:30pm, -yes, i'm a late nite thinker not as David-) just to build a widget to select n-choices from a list... geez, that IDL handbook is cryptic indeed (japanese if far easier to understand), and although it works i'm pretty sure it could be far easier implemented. And at the very end it shows that it wasn't that much difficult, but it was a pain trying to figure out the way widgets work from the lousy examples... (no use trying to read the text: nowhere is explained what 'pro widget2_event, ev' is and even less what that ev structure is and means. The example shows how to build a list, but is never shown how to retrieve the values chosen from that list -that's what took me most of the time-).

And to call it a day, a college just happend to ask me to give her a quick and dirty explanation of what an object is...

All of this just says one thing: Objects are easy to use once you know how. Objects are useful. All we need is to learn how to use them and that's

difficult by now.

So I hope RSI provides with the next release with an 'IDL Objects for old farts dumb scientists'.

In any case, I support that proposal for Pavel and 'Now, what's an object again?' Ben to contribute to David's -never in the library's shelf- book.

And now i better hurry home... otherwhise i'd spend the night at the lab.

Good night, and 'don't let the OG bugs bite you'

Luis Alonso

PD: is that the right expression 'good night and don't let the bed bugs bite you'? My english's expressions are not in shape... i keep on shuffling the words :\

Subject: Re: Top 10 for old farts
Posted by Michael Cugley on Thu, 03 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Martin Schultz <martin.schultz@dkrz.de> writes:

> Hmm, you could probably have saved at least four hours had you gone
> through David's widget examples instead of reading the manuals. At
> least, this is how I learned to deal with these things (admittedly, I
> had to look twice before I understood anything at all).

Personally I found widget programming, with assistance from HyperHelp to be fairly easy. But then maybe that's because I've programmed in Visual Basic and X before...
--
Michael Cugley (mjcugley@medphys.dundee.ac.uk)

Subject: Re: Top 10 for old farts
Posted by Martin Schultz on Thu, 03 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Luis Alonso wrote:
>
>
> Today i've spent a whole afternoon (from 3pm to 21:30pm, -yes, i'm a late
> nite thinker not as David-) just to build a widget to select n-choices from
> a list... geez, that IDL handbook is cryptic indeed (japanese if far easier

> to understand), and although it works i'm pretty sure it could be far easier
> implemented. And at the very end it shows that it wasn't that much
> difficult, but it was a pain trying to figure out the way widgets work from
> the lousy examples... (no use trying to read the text: nowhere is explained
> what 'pro widget2_event, ev' is and even less what that ev structure is and
> means. The example shows how to build a list, but is never shown how to
> retrieve the values chosen from that list -that's what took me most of the
> time-).
>

Hmm, you could probably have saved at least four hours had you gone
through David's widget examples instead of reading the manuals. At
least, this is how I learned to deal with these things (admittedly, I
had to look twice before I understood anything at all).

Martin

--
[[[[[[[[[[
[[Dr. Martin Schultz Max-Planck-Institut fuer Meteorologie [[
[[Bundesstr. 55, 20146 Hamburg [[
[[phone: +49 40 41173-308 [[
[[fax: +49 40 41173-298 [[
[[martin.schultz@dkrz.de [[
[[[[[[[[[[