
Subject: Top 10 IDL Requests

Posted by [davidf](#) on Sun, 16 Jul 2000 07:00:00 GMT

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

Hi Folks,

I was teaching an IDL course at RSI this past week and had the opportunity to do a bit of intelligence work on behalf of this newsgroup at the annual RSI picnic on Thursday. Besides the usual volleyball and Frisbee, some of us sat around and talked shop. Sigh...

I'll get a life later, but I thought you might like to know the IDL 5.4 beta is do to go out in the next week or so and they are signing up testers. If you would like to participate, you can contact the IDL beta program at idl_beta@rsinc.com.

I would especially encourage those of you who want more math functionality in IDL to do so, because that seems to be one of the themes of this release. Apparently the new math guy they hired recently has been busy. I think Craig in particular should have a look at the new curve fitting routines.

There are other things that should warm the heart of any regular IDL newsgroup reader. For example, a completely rewritten SPAWN command for Windows will be warmly embraced. Not only can you now get the results of the spawned command back (the way you have always been able to in UNIX), but you can also use the NOSHELL keyword to avoid the ugly DOS window popping up all the time. And J.D. will appreciate the new relaxed structure rules that will allow anonymous structures (with similar fields) to be concatenated.

And, as often happens, the biggest changes under the hood get short--shrift in the marketing hype. But the fact that the Windows version of IDL now supports LARGE data files meant a four-month effort to work-around Microsoft's very restricted I/O layer. But this will be a godsend for those of you working with those 40 GByte image files. :-)

One of the people I ran into at the picnic was the new VP of Software Engineering. After we were introduced and it was pointed out that I was, uh, the person who wrote "that book about IDL" the VP gave me his e-mail

and phone extension and encouraged me to submit my top 10 IDL requests to him personally.

Wow! You don't get an opportunity like this every day. :-)

Not wanting to hog the spotlight, however, I thought I'd have a little contest and see what the rest of you want. I've already reserved the top spot for taking the Table Widget out in the field behind RSI and shooting it, but I'm open to suggestions after that.

I do feel strongly that the TV command could be made a whole lot smarter. I find I can't even use it anymore if I want to write device/version/decomposed_state independent code. Certainly you can use my TVIMAGE or Liam's IMDISP program, but shouldn't something as simple as this be built into IDL? And please, God, don't EVER take my 24-bit image though a color table, no matter what silly thing I've done to the color decomposition state. :-(

(One thing that seems obvious from reading the list of new features in IDL 5.4 is that in today's employment marketplace it is futile to assign good engineers to fixing old technology. They will just go down the street and work for someone else doing something a whole lot sexier for twice as much money. It's reality I guess, but it seems a shame too. RSI is certainly not the only software company suffering from the current state of affairs.)

So, here is the deal. If you feel inclined to submit a couple of ideas to this newsgroup, I will collect them and submit them personally to the VP. I'll even follow up and make sure he has, uh, read them. :-)

Please restrict yourself to a paragraph or so about why you think your suggestion is a good idea. Suggestions can either be fixes to something currently in IDL, or requests for future capabilities. Items that make it to the top of the list will be written with some wit, intelligence, and humor. Items at the bottom of the list will be written with a lot of whining. I reserve the right to edit submissions if I think it is unintelligible to the average computer dolt.

That's it! No prizes or anything. Just a chance to be heard by someone who can maybe make a difference. In today's world, that's about all we can ask. :-)

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 IDL Requests
Posted by [Harvey Rarback](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt wrote:

>
> "richard hilton" <rdh5@dmu.ac.uk> writes:
>
>> How about an equivalent of the C-commands CONTINUE and BREAK to control
>> loops. This would be more elegant than the "GOTO....label" that exists at
>> the moment. Maybe I'm being pedantic but this seems an obvious change.
>
> I like this suggestion a lot!
>
> Oooh, I just thought of something. This may cause name conflicts.
> People who use CONTINUE or BREAK as variable names would not be happy
> if the language suddenly changed. Still, I would highly prize these
> constructs if they could be implemented discretely.

How about implementing this with another argument to COMPILE_OPT?

BTW, I love the behavior of COMPILE_OPT IDL2 and use it in all my new code. No more putting those dumb "L's" after all my integer constants.

--Harvey

Subject: Re: Top 10 IDL Requests
Posted by [Phillip David](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'd agree with a way to get 'at home' licenses. I can't connect with the license server at work because it's not available over the internet (it's on a separate, classified network).

H C Pumphrey wrote:

- > (2) Free or cheap 'at home' licenses for people who use IDL at work. I'm
 - > gradually abandoning IDL because my boss won't cough up UKP 1000 or so
 - > for me to have IDL on my home machine
-

Subject: Re: Top 10 IDL Requests

Posted by [Craig Markwardt](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

"richard hilton" <rdh5@dmu.ac.uk> writes:

- > How about an equivalent of the C-commands CONTINUE and BREAK to control
- > loops. This would be more elegant than the "GOTO....label" that exists at
- > the moment. Maybe I'm being pedantic but this seems an obvious change.

I like this suggestion a lot!

Oooh, I just thought of something. This may cause name conflicts. People who use CONTINUE or BREAK as variable names would not be happy if the language suddenly changed. Still, I would highly prize these constructs if they could be implemented discretely.

Craig

--

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

Subject: Re: Top 10 IDL Requests

Posted by [Craig Markwardt](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

landsman@my-deja.com writes:

- > In article <on66q4a433.fsf@cow.physics.wisc.edu>,
- > craigmnet@cow.physics.wisc.edu wrote:
- >
- >> * A way to index strings like arrays.

```
>>
>> I know we can use STRMID and STRPUT, but it seems that an array-like
>> notation would fit so much better with the philosophy of IDL.
>>
>
...deletia...
>
> But I am not sure that such a change would be desirable. First of
> all would be the difficulty in making a syntax change backwards
> compatible. The change of the string syntax from V1 to V2 probably
> required more coding changes than any other syntax change ever made to
> IDL. Also, some syntax would become more complicated -- for example, the
> current extraction of elements of a string array st1 = st[3:5] would
> have to become st1 = st[* ,3:5]. V5.3 introduced many string
> processing enhancements (e.g. STRMID now accepts vector parameters), so
> I think all the functionality of the array notation is now available in
> then string processing functions. Finally, one can always index strings
> like arrays by first converting them with BYTE().
```

I agree that standard array indexing syntax wouldn't mix too well with accessing strings. On the other hand we do already have the X.(SUBSCRIPT) notation for structures, which could perhaps be adapted for strings as well.

```
IDL> value = '0123456789'
IDL> value.(4:5) = '--' ;; pre-beta!
IDL> print, value
0123--6789
```

This is intuitive to me at some level. You use the "." notation to get at the internals of a structure. The same notation could get at the internals of a string.

For that matter, it would be nice, but probably hard to implement, to be able to use range and "*" notation for structures in the same way.

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [promashkin](#) on Tue, 18 Jul 2000 07:00:00 GMT

Thanks for the initiative David, and People, cool down your jets! This thread is insane. Christmas is too far ahead yet, and not even Santa will be able to fulfill a third of our requests. This way, you'll kill David's desire to help out, and that poor VP will not ever again offer to read suggestions. We all want improvements in our own particular way, but I'd rather see one change made to IDL in a new release, than have a comprehensive list of wishes sent to RSI, which will never implement them all.

Cheers,
Pavel

Subject: Re: Top 10 IDL Requests

Posted by [Liam E. Gumley](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

"Kenneth P. Bowman" wrote:

>
> In article <3972CF0F.142752CD@mpipsykl.mpg.de>, Benno Puetz
> <puetz@mpipsykl.mpg.de> wrote:
>
>> Color support in PS output (for PLOT and the like).
>
> I wholeheartedly second that one.
>
> Liam suggests using the "Printer" device, but that is YATTFO (Yet
> Another Thing To Figure Out). "Real" 24-bit color in the PS device is
> simply the right thing to do. :-)

If decomposed color can be supported on color PostScript printers via the PRINTER device, then one suspects it should be possible to support decomposed color directly with the PS device.

> Since Adobe is moving to PDF as the Illustrator filetype, how about
> direct to PDF?

I'd settle for encapsulated PostScript output with a decent preview image that could be imported into Word.

Cheers,
Liam.

Subject: Re: Top 10 IDL Requests

Posted by [Struan Gray](#) on Tue, 18 Jul 2000 07:00:00 GMT

Has anyone else noticed that despite David's assurance that no hip programmer would be seen dead fixing old code, that's exactly what all of us want RSI to do? They probably hope we're all locked into legacy code

In my group at least one of the major reasons people haven't been persuaded to buy IDL has been the need to learn a whole new bag of workarounds. Reliable, predictable keyword use and mathematical accuracy may not be sexy, but it's something people actually want.

Looking at the Matlab evil empire (Boo Hiss) and my favourite quality graphing program (Igor) one other thing IDL doesn't offer is easy interfacing to data acquisition hardware or the web. For sophisticated control you need to get your hands dirty with C++ or ION\$\$\$\$, but it's amazing how much can be achieved with a few simple vendor-provided hooks for text or raw binary io. We just interfaced an old VME crate to Igor on the Mac using strings dumped into text files and Personal Web Sharing. Took a morning, and most of that was reading the VME ethernet card manual.

Struan

Subject: Re: Top 10 IDL Requests

Posted by [Martin Schultz](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

David Fanning wrote:

>
>
> One of the people I ran into at the picnic was the
> new VP of Software Engineering. After we were introduced
> and it was pointed out that I was, uh, the person who
> wrote "that book about IDL" the VP gave me his e-mail
> and phone extension and encouraged me to submit my
> top 10 IDL requests to him personally.
>
> Wow! You don't get an opportunity like this every day. :-)
>
>
> Cheers,
>
> David

Try to keep it short for there are sooo many wishes (BTW: what does

this tell us?)

- (1) multithreading
- (2) complement keyword to where
- (3) end of array syntax (I can't see why A should not be feasible)
- (4) working contour routine and map projections (simple example:
IDL> map_set,limit=[-60,-180,60,360]
IDL>
map_continents
should give America twice, I think ...
- (4a) ... and more sophisticated map labels, e.g. 60 (degreesymbol) N instead of
just 60 as an option.
- (5) improved color handling for TV and TVRD
- (6) adapt a colorbar and a legend procedure (e.g. from the ESRI library). This is
something basic enough that it should be maintained by RSI - even though everyone
will be free to use something different
- (7) an "IN" operator, e.g. selected = where(species in ['Ozone', 'NOx', 'CO'])
- (8) higher level graphic objects, e.g. extending your XPLOT, ****and**** a contour object
and map support! The high level plot object should allow things like:
XPlot = obj_new('Xplot',x=x,y=sin(x)) ; generate and show object
XPlot->Add,x=x2,y=cos(x2) ; add another data set
XPlot->SetProperty,curve=2,psym=8,line=2 ; modify properties of second curve
etc.

Since I start dreaming again, I better stop here.

Cheers (from wonderful but cloudy Germany),
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 IDL Requests
Posted by [Bruce Bowler](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

HTML help? Ickkk...

On the subject of help, I was reading one of the paper manuals (don't recall which one) a few weeks ago (it's just not the same taking a computer to "the reading room") and stumbled across a line that almost made me fall off "the chair". It said (paraphrasing since I can't find the original page, and, remember this is the PAPER copy of the documentation) "click here for more information about 'x'".

Someone else somewhere else mentioned that they thought the RT license should be cheaper... Let me add another "business practices" issue to the top 10. If RSI wants to gain inroads into big (or little) business, they should GIVE AWAY the student edition (some would say that \$79 is "giving it away" but I would (obviously) disagree). Oh yeah, keep it up to date, the web site says the current version is based on 5.0, why isn't the current SV based on 5.3?

My (as an IDL novice) number 1 issue would be to "clean up the syntax". Some specific suggestions (I "grew up" using FORTRAN and VMS if that helps explain my position at all)

- get rid of the comma that immediately follows a command
- it ain't intuitive that /argument is equivalent to argument=1, I prefer the latter.

I'm sure people who have been around IDL longer than I have probably have more things they dislike about the syntax...

I guess that's enough for now...

wmc@bas.ac.uk wrote:

```
>
> Luis Alonso <luis.alonso@uv.es> wrote:
>> I use IDL's help every day.
>
>>> IDL Online Help facility, and I heartily believe that this help
>>> facility is one of IDL's strongest and most oft-neglected points.
>
> !?! One of my requests would be to throw away the current help system
> and provide an html-based version.
>
> The other big request would be to make "contour" work properly
> so that /cell_fill is unnecessary. I mean, in this day and age...
>
```

> -W.
>
> ps - sorry no wit...
>
> --
> William M Connolley | wmc@bas.ac.uk | <http://www.nerc-bas.ac.uk/icd/wmc/>
> Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself

Subject: Re: Top 10 IDL Requests
Posted by [Larry Ashim](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Why think small?

IDL should have a true WYSIWYG for graphics. Something that allows me to click on an axis displayed on the screen and change the range, font size, etc. Then let me click on the graphic, copy it, and paste it into another application. This would make the preparation of presentations and reports with IDL graphics 100% easier. Sure PostScript produces beautiful printed output but most of the graphics I need are written to the screen, either directly or through another application program.

If EXCEL, PSI Plot, Dplot, etc., can do it, why can't IDL. And Dplot is free, not \$1700 or so.

Larry

Subject: Re: Top 10 IDL Requests
Posted by [Richard Tyc](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

My two bits would be increased performance for Object graphics especially volume objects which do not make use of any OpenGL hardware acceleration so volume rendering speeds are quite slow even on multi CPU machines. This would greatly help in developing medical imaging apps where volume rendering MRI/CT data sets is becoming quite common.

Rich

Subject: Re: Top 10 IDL Requests
Posted by [Richard Tyc](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

anne martel <anne.martel@nottingham.ac.uk> wrote in message
news:39745A3A.CECAC2FA@nottingham.ac.ukqqq...

> My main request is for the ability to make polygon objects transparent.
> Someone posted a request for information on how to do this a short while
> ago but I don't think anyone responded so I'm assuming that it's not
> possible at the moment (unless you use volume rendering which is too
> slow).

If you texture map a RGB+alpha IDLgrImage object into the polygon object,
you can gain transparency assuming you set the alpha channel low enough and
the appropriate blend function (I use [3,4]) to IDLgrImage.

Subject: Re: Top 10 IDL Requests

Posted by [anne](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

My main request is for the ability to make polygon objects transparent.
Someone posted a request for information on how to do this a short while
ago but I don't think anyone responded so I'm assuming that it's not
possible at the moment (unless you use volume rendering which is too
slow).

I would also like to vote for fixing the bug that turns single element
arrays into scalars.

Anne Martel
Dept Medical Physics
QMC, Nottingham
<http://www.nottingham.ac.uk/radiology>

Subject: Re: Top 10 IDL Requests

Posted by [wmc](#) on Tue, 18 Jul 2000 07:00:00 GMT

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

Luis Alonso <luis.alonso@uv.es> wrote:

> I use IDL's help every day.

>> IDL Online Help facility, and I heartily believe that this help
>> facility is one of IDL's strongest and most oft-neglected points.

!?! One of my requests would be to throw away the current help system
and provide an html-based version.

The other big request would be to make "contour" work properly
so that /cell_fill is unnecessary. I mean, in this day and age...

-W.

ps - sorry no wit...

--

William M Connolley | wmc@bas.ac.uk | <http://www.nerc-bas.ac.uk/icd/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself

Subject: Re: Top 10 IDL Requests
Posted by [Luis Alonso](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here we go!

I use IDL's help every day.
It's annoying trying to find commands or operators and finding no reference to them or just a single sentence, most of the time very cryptic.
I'm still working with IDL5.0 so maybe some things have already changed, but i thing it'd pay to rewrite most of the help from scratch in a more comprehensible way. And with FULL examples using at least the main Keywords of each procedure/fuction. As everybody knows after using IDL a couple of hours keywords 'make the difference', but funny that the help one obtains about them is practically unexistent.

So, that's one of my needs.

I'll read the rest of the proposals, and see if i have something to add ;)

This is a great opportunity!!!!

"David Bowman" <bowman@ipgp.jussieu.fr> wrote in message
news:170720001448597688%bowman@ipgp.jussieu.fr...

>

>

> What an amazingly welcome opportunity to air petty grievances! Many
> heartfelt thanks both to you and the anonymous VP...

>

> I must admit that I am not a code-god like some of the frequent posters
> to this newsgroup. :-) As a mere computational mortal, I frequent the
> IDL Online Help facility, and I heartily believe that this help
> facility is one of IDL's strongest and most oft-neglected points.

Subject: Re: Top 10 IDL Requests
Posted by [Olivier ARCHER](#) on Tue, 18 Jul 2000 07:00:00 GMT

My suggests:

- emacs like bindings in idlhelp (ctrl-a ctrl-k ...)
- multi window support: when you create many windows whith differents axis / map_set, IDL forgot the system projection so you have to save by hand all !X !Y, !MAP, !MOUSE...etc. It should be nice if IDL manage it internaly.
- same remarks for !P.multi. You can't go back a previous plot without losing axis.

--

Olivier ARCHER
IFREMER DRO OS

Subject: Re: Top 10 IDL Requests
Posted by [hcp](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I just thought I'd add my 2-penn'orth here.

- (1) arrays of zero size. (Someone has already explained why.)
- (2) Free or cheap 'at home' licenses for people who use IDL at work. I'm gradually abandoning IDL because my boss won't cough up UKP 1000 or so for me to have IDL on my home machine [1]
- (3) More reliable support for different colour depths in X. Yorick and R do the right thing in 16-bit displays without you having to even think about it. Why doesn't this happen in a package you pay 1000 quid for?
- (4) make sure support for Unix (including Linux) is maintained at the same level as support for WinDoze.
- (5) Er
- (6) Thats it.

Hugh

[1] No, I can't connect to the license server at work. We don't have free local phone calls in the Yookay

--

=====
=====

Hugh C. Pumphrey | Telephone 0131-650-6026
Department of Meteorology | FAX 0131-650-5780
The University of Edinburgh | Replace 0131 with +44-131 if outside U.K.
EDINBURGH EH9 3JZ, Scotland | Email hcp@met.ed.ac.uk
OBDisclaimer: The views expressed herein are mine, not those of UofE.

=====
=====

Subject: Re: Top 10 IDL Requests
Posted by [richard hilton](#) on Tue, 18 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

How about an equivalent of the C-commands CONTINUE and BREAK to control loops. This would be more elegant than the "GOTO....label" that exists at the moment. Maybe I'm being pedantic but this seems an obvious change.

Rich

Subject: Re: Top 10 IDL Requests
Posted by [promashkin](#) on Wed, 19 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

> wmc@bas.ac.uk wrote:

>

> If so, could this be extended to allow *any* structures to be put into
> arrays? Is there any particular reason why all array elements have to
> be the same type of structure?

Let's allow putting into the same array the floats, strings and object references, too. Why not. I bet RSI developers will then certainly have a fit. So far they used to recommend pointer arrays for such mixtures, but we can always use another way to write code. All we'd ever need would be "ANYTYPEARR(x)" and "ANYTYPEINDGEN(x)" ...

> ps - while I'm here - can replicate be made to work on arrays?

It would be somewhat redundant with REBIN, but sure, why not.

Cheers,
Pavel

Subject: Re: Top 10 IDL Requests
Posted by [wmc](#) on Wed, 19 Jul 2000 07:00:00 GMT

David Fanning <davidf@dfanning.com> wrote:

> And J.D. will appreciate the new relaxed structure rules
> that will allow anonymous structures (with similar fields)
> to be concatenated.

You mean "be put into arrays together", not concatenated, don't you?

If so, could this be extended to allow **any** structures to be put into arrays? Is there any particular reason why all array elements have to be the same type of structure?

-W.

ps - while I'm here - can replicate be made to work on arrays?

--

William M Connolley | wmc@bas.ac.uk | <http://www.nerc-bas.ac.uk/icd/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself

Subject: Re: Top 10 IDL Requests

Posted by [kashyap](#) on Wed, 19 Jul 2000 07:00:00 GMT

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

1. CALL_EXTERNAL

No one I know has come close to implementing a C or Fortran `call_external` in Solaris 2.5+ -- perhaps it's simply a matter of providing better documentation.

2. _EXTRA

Please consider having all built-in commands accept `_EXTRA` as a keyword.
pro: making wrapping routines will be a cinch
con: must pay attention to keyword names across the entire suite of programs

3. WHERE

This has been mentioned before, but doesn't hurt to repeat -- a way to extract the complement of a `WHERE()` output.

4. last element

Also has been mentioned before. `A` means to index the last element of an array. Since `"*` has a lot of baggage, perhaps some other symbol, say `"%` can be used -- i.e.,

A[%] \equiv A[N_ELEMENTS(A)-1L]

5. For all built-in functions and procedures

If called with no arguments, or with fewer arguments than required, please print out the complete (keywords and all) calling sequence.

6. Debugger

Variable Watch interface could use a lot of user-friendliness. For example, that spiffy-looking tabular listing should go away. The display is too tiny, it doesn't remember scroll position from step to step, arrays are a pain to look through, etc.

--

kashyap@head-cfa.harvard.edu

617 495 7173 [CfA/P-146] 617 496 7173 [F]

Subject: Re: Top 10 IDL Requests

Posted by [Jeff Guerber](#) on Fri, 21 Jul 2000 07:00:00 GMT

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

Wow, it looks like we're finally able to post again! And it only took them four months!

Anyway, after thinking about it for a while, here's my IDL wishlist. Some of the things have been mentioned by other people, but I'm including them anyway to add my vote. Beyond the first few, they're not necessarily in any particular order of preference.

1) Definable lower array bounds. I often have to translate algorithms originally developed in Fortran, and changing everything to 0-based indexing is a MAJOR pain (not to mention bug-prone). Besides, my fingers start at "one", not "zero"! :-) In some cases I'd like to go from, say, -10:+10; yeah, I know I can use an extra array, but what about -10000:+10000 ? Fortran's had this for DECADES... why not IDL?

2) It's probably waaaay too late to change this, but: I find it really annoying that pointer dereferencing is a low-precedence prefix operator; I'd much prefer a postfix operator at the same precedence as structure-field references and array subscripts. So instead of having to write, for example,

`(*statep).plotIDSp)[i]`

(which I actually took from one of my programs), I could simply say,

`statep*.plotIDSp*[i]`

which IMHO is MUCH more readable -- no excess parentheses and you don't have to jump clear back to the front of the expression to see that "plotIDSp" is a dereferenced pointer. Even "(statep*).(plotIDSp*[i])" would be better than the current form. For dereferencing an array of pointers, the syntax would be "array[i]*".

3) Class data, public data, and private methods in objects.

4) Initialization in structure definitions. I'd like to be able to write, for example,

```
pro struct__define
void = { struct, a:2, b:4.5, c:"Fanning's book is great" }
return
end
```

and have a new {struct} automatically be initialized to {a:2, b:4.5, c:"Fanning's book is great"} instead of {a:0, b:0.0, c:""}.

5) Specifying colors by name... preferably the whole X11 palette from /etc/X11/rgb.txt. I've sometimes toyed with the idea of writing an object that would read rgb.txt, load colors _on demand_... and keep track of where in the color table each went, so that any one color doesn't get loaded into multiple table entries.

6) A standard, and _comprehensive_, I/O package for FITS files. Yes, I know there are about 5 packages available in Wayne's (excellent!) idlastro library; but they all have different capabilities, and until recently there was no good way of figuring out which was best for an application. (I think there's now a document that discusses the strengths and weaknesses of each package.) Given IDL's background and popularity in astronomy, I've always been puzzled by its lack of built-in FITS support.

7) Recalling the graphics system variables on a window-by-window basis, so we don't have to restore them manually when changing windows.

8) String subscripting.

9) BREAK and CONTINUE (or CYCLE) statements for loops.

10) Complement of WHERE.

11) Subscript shorthand for the upper-bound array index.

Sorry, I guess that's more than 10, isn't it? Oh well, some of them probably don't have the proverbial snowball's chance of getting into the

language anyway!

Jeff Guerber

Raytheon ITSS
NASA Goddard Space Flight Center
Oceans & Ice Branch (code 971)

Any opinions here are my own. Well, in a few cases I borrowed them from other people, but they're certainly not Raytheon's or NASA's!

Subject: Re: Top 10 IDL Requests

Posted by [robert.m.candey.1\[2\]](#) on Fri, 21 Jul 2000 07:00:00 GMT

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

My biggest wish is really mundane (but we've been asking for this for forever): labeling minor ticks on log plot axes when less than 2 decades are shown (and maybe labeling them on request anytime). Often there is no info what the axis range is for 0 or 1 major ticks.

Also mundane but useful would be such things as:
Better grouping of structures in the system variables, like having all text info together (text, font color, size, orientation, font type, etc.)

At least !x, !y, !z should add titlecolor, label color and size

I believe there was a nice wish list discussion on this group last May that might be worth reviewing.

--

Robert.M.Candey@gsfc.nasa.gov 1-301-286-6707 (286-1771 fax)
NASA Goddard Space Flight Center, Code 632
Greenbelt MD 20771 USA <<http://nssdc.gsfc.nasa.gov/personnel/rcandey/>>

Subject: Re: Top 10 IDL Requests

Posted by [noymer](#) on Fri, 21 Jul 2000 07:00:00 GMT

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

In article <39749303.3E1B9FEE@ssec.wisc.edu>,
"Liam E.Gumley" <Liam.Gumley@ssec.wisc.edu> wrote:

>
> I'd settle for encapsulated PostScript output with a decent preview
> image that could be imported into Word.
>

You mean EPS where the decent preview image is *OPTIONAL*, right Liam?

I've got nothing against it per se, but it makes the EPS files both bigger and harder to compress, and I never need it anyway (don't use Word). If it is optional, then those of you who need it can use it all you want and those of us who don't can have smaller output files. Everyone is happy.

- Andrew

Sent via Deja.com <http://www.deja.com/>
Before you buy.

Subject: Re: Top 10 IDL Requests
Posted by [promashkin](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Learning just about anything, not just IDL syntax, is annoying if you don't want to do it. Why not offer them at RSI to just sell their company, forget developing IDL and go look for jobs as C programmers? Forget Perl too, C is okay for everything :-(

Let's just say I hope Eric's request does not make it to the David's top ten list.

Cheers,
Pavel

Among other things, Eric Vella wrote:

> ... Rewrite IDL as a Perl module. The IDL
> syntax is really annoying, with all sorts of hard to remember quirks and
> strange behavior everywhere ...
> ... standard language like Perl, that would be
> enough incentive for me to learn the language and become an advanced
> user

Subject: Re: Top 10 IDL Requests
Posted by [Craig Markwardt](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Eric Vella <eric.vella@kla-tencor.com> writes:

> ... But if
> IDL could leverage a useful standard language like Perl, that would be

> enough incentive for me to learn the language and become an advanced
> user. In addition, IDL would instantly and automatically pick up an
> enormous code base of functions, including all sorts of useful numerical
> algorithms.

In that case, PDL, the Perl Data Language, may suit your needs. As far as I understand, it is quite distinct from IDL in terms of syntax, but has the same concept of vectorized computation.

<http://www.aao.gov.au/local/www/kgb/perldl/>

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [Eric Vella](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Here's a more ambitious request, which I don't expect RSI to consider, but what the heck: Rewrite IDL as a Perl module. The IDL syntax is really annoying, with all sorts of hard to remember quirks and strange behavior everywhere. This was not so bad when IDL was young and had just a few simple commands, but in recent years RSI has expanded IDL into a full-blown language. This is a classic example of reinventing the wheel, badly. I have no interest in learning a complete new computer language, so I will never be an advanced IDL user -- I will just use direct graphics and the small subset of IDL I am familiar with. But if IDL could leverage a useful standard language like Perl, that would be enough incentive for me to learn the language and become an advanced user. In addition, IDL would instantly and automatically pick up an enormous code base of functions, including all sorts of useful numerical algorithms.

Subject: Re: Top 10 IDL Requests
Posted by [Michael Asten](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

My top few requests:

1) Try harder to keep backward compatibility when releasing new

versions, eg how silly when the TODAY function disappeared on transition from idl5.2 to 5.3 (problem with y2k compliance apparently).

- 2) Add more features to smooth the development path for the non-professional programmer (ie the scientific user).
Examples would be to add a hardcopy print button (instead of burying the info on your website), introduce more canned
guis (dialog_message and dialog_pickfile are great - why not a few standard guis to
a) create a plot window on a widget,
b) to provide a template of labelled fields for input/editting of parameters (a userfriendly cleanup of xvaredit)
(see thread on IDL-pvwave newsgroup)
c) a thermometer gui to show progress during a slow looping task
- 3) a gui to take " standard output" so that info normally sent to the log window can be displayed when the logwindow is absent (eg when a program is run under runtime licence).

4) I'll add a vote for plot colors accessible by name.

5) a check at compile time for multiple routines of the same name (programmer error creating routines of same name but different function, is unchecked at present, but obviously causes unpredictable results).

6) check on validity of loop indices, eg
for i=0,10,0

will hang idl on a PC, so that only control-alt-delete can stop it. If the indices are computed in error (eg with roundoff or wrong sign) then location of the error is difficult since the infinite loop prevents any break from the keyboard on a single threaded system.

Given that rsi may not want to make loop execution any slower, this option could perhaps be a compile-time option for desperate debug purposes only.

Regards,
Michael Asten

Subject: Re: Top 10 IDL Requests
Posted by [Michael Asten](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

>
>
> Items that make it to the top of the list
> will be written with some wit, intelligence,
> and humor.

I guess this is why you are calling for us other users to write in,
eh Dave ? :=)

Regards,
Michael Asten

PS Many thanks David for taking this on. I will send in my \$0.02,
especially on the need for an upgraded widget generator including
compound widgets, which I'm sure you will support with impeccably
unbiased objectivity.

Subject: Re: Top 10 IDL Requests
Posted by [Michael Asten](#) on Fri, 21 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

>
>
> Items that make it to the top of the list
> will be written with some wit, intelligence,
> and humor.

> I guess this is why you are calling for us other users to write in ?
> :=)

Regards,
Michael Asten

PS Many thanks David for taking this on. I will send in my \$0.02,
especially on the need for an upgraded widget generator including
compound widgets, which I'm sure you will support with impeccably
unbiased objectivity.

Subject: Re: Top 10 IDL Requests
Posted by [dsreyn](#) on Mon, 24 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I would find it extremely helpful if it were possible to extend COMMON blocks,
without being forced to exit and restart IDL.

Doug

Subject: Re: Top 10 IDL Requests
Posted by [kashyap](#) on Tue, 25 Jul 2000 07:00:00 GMT

Ah, the specific reason I came up with that request was that I was trying to have CURVEFIT pass strange keywords down to a curve-defining function (which is called via CALL_PROCEDURE), and failing. In that particular case, the fix is simple of course (edit CURVEFIT).

Also, I was not aware that the function definition does not **have** to have the "_extra" -- not quite what the help file says:

You can pass keyword parameters to called routines by value by adding the formal keyword parameter "_EXTRA" (note the underscore character) to the definition of your routine. ...

now that I am primed to it, I notice that it goes on to say:

When the keyword _EXTRA appears in a procedure or function call, its argument is either a structure containing additional keyword/value pairs which are inserted into the argument list ...

It's not tremendously clear that putting the _extra in a function definition is optional. I suppose it helps only if the function in question calls other functions or subroutines to which it must pass keywords.

Thanks for the clarifications,
Vinay

In article <on1z0hrdn4.fsf@cow.physics.wisc.edu>,
Craig Markwardt <craigmnet@cow.physics.wisc.edu> wrote:
> davidf@dfanning.com (David Fanning) writes:
>
>> Craig Markwardt (craigmnet@cow.physics.wisc.edu) writes:
>>
>>> davidf@dfanning.com (David Fanning) writes:
>>>> Vinay L. Kashyap (kashyap@head-cfa.harvard.edu) writes:
>>>> > 2. _EXTRA
>>>> >
>>>> > Please consider having all built-in commands accept _EXTRA as a keyword.
>>>>
>>>> Uh, this is the way it works. :-)
>>>
>>> Uh, not quite. There are some built in commands that don't accept any
>>> keywords at all. The _EXTRA keyword doesn't work for them, **even** if
>>> the value passed is empty!
>>>
>>> Why is this important? Makes it a pain to write a wrapper procedure

```

>>> or function.
>>
>> Alright, I must be obtuse today, but I can't figure out why
>> it would be hard to write wrapper routines for commands that
>> don't take keywords. Surely in writing the wrapper you give
>> at least *some* thought to what keywords you might expect
>> to be passed. Adding an _Extra to such a command seems
>> excessively anal at the very least, and certainly unnecessary. :-)
>>
>> And what commands did you have in mind? I've never encountered
>> a built-in command that didn't accept this keyword mechanism.
>
> Hrrm. The moment I need to find an example, and I can't find it.
> Arghh. An example of a built-in command that doesn't take keywords is
> EMPTY, but I agree that it's a pretty lame example. I actually would
> hope that *all* procedures and functions could be called with _EXTRA,
> whether or not they actually accept keywords. For, example, this
> statement
>
> call_procedure, 'EMPTY', _EXTRA=null
>
> will fail no matter what, even if null is an undefined variable.
> Shouldn't IDL be smart enough to test whether the _EXTRA value is
> undefined before it crashes?
>
> I'm always looking for ways to avoid special cases in wrapper
> routines. Real world examples of such unavoidable abominations are
> given below.
>
> Craig
>
>
> (from XFWINDOW in XFWINDOW_CALL_PROCEDURE)
> sz = size(key)
> if sz(sz(0)+1) EQ 8 then begin ;; Keywords are present
>   xfwindow_rekey, key
>   case n_args of
>     0: call_procedure, cmd, _extra=key
>     1: call_procedure, cmd, x0, _extra=key
>     2: call_procedure, cmd, x0, x1, _extra=key
>     3: call_procedure, cmd, x0, x1, x2, _extra=key
>     4: call_procedure, cmd, x0, x1, x2, x3, _extra=key
>     5: call_procedure, cmd, x0, x1, x2, x3, x4, _extra=key
>   endcase
> endif else begin           ;; No keywords are present
>   case n_args of
>     0: call_procedure, cmd
>     1: call_procedure, cmd, x0

```



```

> 2: call_procedure, cmd, x0, x1
> 3: call_procedure, cmd, x0, x1, x2
> 4: call_procedure, cmd, x0, x1, x2, x3
> 5: call_procedure, cmd, x0, x1, x2, x3, x4
> endcase
> endelse
>
> (from MPFIT in MPFIT_CALL)
> if proc then begin
>   if n_params() EQ 3 then begin
>     if n_elements(extra) GT 0 then $
>       call_procedure, fcn, x, f, fjac, _EXTRA=extra $
>     else $
>       call_procedure, fcn, x, f, fjac
>     endif else begin
>       if n_elements(extra) GT 0 then $
>         call_procedure, fcn, x, f, _EXTRA=extra $
>       else $
>         call_procedure, fcn, x, f
>       endelse
>     endif else begin
>       if n_params() EQ 3 then begin
>         if n_elements(extra) GT 0 then $
>           f = call_function(fcn, x, fjac, _EXTRA=extra) $
>         else $
>           f = call_function(fcn, x, fjac)
>         endif else begin
>           if n_elements(extra) GT 0 then $
>             f = call_function(fcn, x, _EXTRA=extra) $
>           else $
>             f = call_function(fcn, x)
>           endelse
>         endelse
>       endelse
>     endelse
>   --
> -----
> Craig B. Markwardt, Ph.D.      EMAIL:  craigmnet@cow.physics.wisc.edu
> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> -----

```

--

kashyap@head-cfa.harvard.edu 617 495 7173 [CfA/P-146] 617 496 7173 [F]

Subject: Re: Top 10 IDL Requests

Posted by [Craig Markwardt](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

davidf@dfanning.com (David Fanning) writes:

> Craig Markwardt (craigmnet@cow.physics.wisc.edu) writes:

>

>> davidf@dfanning.com (David Fanning) writes:

>>> Vinay L. Kashyap (kashyap@head-cfa.harvard.edu) writes:

>>>> 2. _EXTRA

>>>>

>>>> Please consider having all built-in commands accept _EXTRA as a keyword.

>>>

>>> Uh, this is the way it works. :-)

>>

>> Uh, not quite. There are some built in commands that don't accept any
>> keywords at all. The _EXTRA keyword doesn't work for them, *even* if
>> the value passed is empty!

>>

>> Why is this important? Makes it a pain to write a wrapper procedure
>> or function.

>

> Alright, I must be obtuse today, but I can't figure out why
> it would be hard to write wrapper routines for commands that
> don't take keywords. Surely in writing the wrapper you give
> at least *some* thought to what keywords you might expect
> to be passed. Adding an _Extra to such a command seems
> excessively anal at the very least, and certainly unnecessary. :-)

>

> And what commands did you have in mind? I've never encountered
> a built-in command that didn't accept this keyword mechanism.

Hrmm. The moment I need to find an example, and I can't find it.

Arghh. An example of a built-in command that doesn't take keywords is
EMPTY, but I agree that it's a pretty lame example. I actually would
hope that *all* procedures and functions could be called with _EXTRA,
whether or not they actually accept keywords. For, example, this
statement

```
call_procedure, 'EMPTY', _EXTRA=null
```

will fail no matter what, even if null is an undefined variable.

Shouldn't IDL be smart enough to test whether the _EXTRA value is
undefined before it crashes?

I'm always looking for ways to avoid special cases in wrapper
routines. Real world examples of such unavoidable abominations are
given below.

Craig

(from XFWINDOW in XFWINDOW_CALL_PROCEDURE)

```
sz = size(key)
if sz(sz(0)+1) EQ 8 then begin ;; Keywords are present
  xfgwindow_rekey, key
  case n_args of
    0: call_procedure, cmd, _extra=key
    1: call_procedure, cmd, x0, _extra=key
    2: call_procedure, cmd, x0, x1, _extra=key
    3: call_procedure, cmd, x0, x1, x2, _extra=key
    4: call_procedure, cmd, x0, x1, x2, x3, _extra=key
    5: call_procedure, cmd, x0, x1, x2, x3, x4, _extra=key
  endcase
endif else begin ;; No keywords are present
  case n_args of
    0: call_procedure, cmd
    1: call_procedure, cmd, x0
    2: call_procedure, cmd, x0, x1
    3: call_procedure, cmd, x0, x1, x2
    4: call_procedure, cmd, x0, x1, x2, x3
    5: call_procedure, cmd, x0, x1, x2, x3, x4
  endcase
endelse
```

(from MPFIT in MPFIT_CALL)

```
if proc then begin
  if n_params() EQ 3 then begin
    if n_elements(extra) GT 0 then $
      call_procedure, fcn, x, f, fjac, _EXTRA=extra $
    else $
      call_procedure, fcn, x, f, fjac
    endif else begin
      if n_elements(extra) GT 0 then $
        call_procedure, fcn, x, f, _EXTRA=extra $
      else $
        call_procedure, fcn, x, f
      endif else begin
        if n_params() EQ 3 then begin
          if n_elements(extra) GT 0 then $
            f = call_function(fcn, x, fjac, _EXTRA=extra) $
          else $
            f = call_function(fcn, x, fjac)
          endif else begin
            if n_elements(extra) GT 0 then $
```

```
f = call_function(fcn, x, _EXTRA=extra) $
else $
f = call_function(fcn, x)
endelse
endelse
```

--

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

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt (craigmnet@cow.physics.wisc.edu) writes:

```
> davidf@dfanning.com (David Fanning) writes:
>> Vinay L. Kashyap (kashyap@head-cfa.harvard.edu) writes:
>>> 2. _EXTRA
>>>
>>> Please consider having all built-in commands accept _EXTRA as a keyword.
>>
>> Uh, this is the way it works. :-)
>
> Uh, not quite. There are some built in commands that don't accept any
> keywords at all. The _EXTRA keyword doesn't work for them, *even* if
> the value passed is empty!
>
> Why is this important? Makes it a pain to write a wrapper procedure
> or function.
```

Alright, I must be obtuse today, but I can't figure out why it would be hard to write wrapper routines for commands that don't take keywords. Surely in writing the wrapper you give at least *some* thought to what keywords you might expect to be passed. Adding an _Extra to such a command seems excessively anal at the very least, and certainly unnecessary. :-)

And what commands did you have in mind? I've never encountered a built-in command that didn't accept this keyword mechanism.

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 IDL Requests
Posted by [wmc](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning <davidf@dfanning.com> wrote:

> William (wmc@bas.ac.uk) writes:

>> If so, could this be extended to allow *any* structures to be put into
>> arrays? Is there any particular reason why all array elements have to
>> be the same type of structure?

> This is the idea behind the changes in IDL 5.4. As long
> as the array elements "match" in the sense of having the
> same amount of storage allocated to them, structures
> can be put into arrays, etc.

Ah, no, that's not enough. I want, say, structures which represent a data agglomeration (I'm trying not to say object) with loads of header fields the same, but a few fields (maybe just the one "data" field) different. So the storage is different. I know I could do this by putting a pointer into the structure instead, but... I can't see why IDL shouldn't do this itself. I guess I'm assuming that, when IDL stores an array of structures, it doesn't store the structures consecutively anyway - just pointers to the structures. In which case, it shouldn't matter what the structure types are. I think.

-W.

ps - I'll hide this here so as to not start a perl/idl war: I use both a lot and I like the syntax of both. But... occasionally the clunkiness of IDL does irritate a little. Like, not allowing null arrays.

--

William M Connolley | wmc@bas.ac.uk | <http://www.nerc-bas.ac.uk/icd/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself

Subject: Re: Top 10 IDL Requests
Posted by [Craig Markwardt](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

davidf@dfanning.com (David Fanning) writes:

> Vinay L. Kashyap (kashyap@head-cfa.harvard.edu) writes:

>> 2. _EXTRA

>>

>> Please consider having all built-in commands accept _EXTRA as a keyword.

>

> Uh, this is the way it works. :-)

Uh, not quite. There are some built in commands that don't accept any keywords at all. The _EXTRA keyword doesn't work for them, *even* if the value passed is empty!

Why is this important? Makes it a pain to write a wrapper procedure or function.

Craig

--

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

Subject: Re: Top 10 IDL Requests

Posted by [bradgom](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

In article <397876AD.283E645D@kla-tencor.com>,

Eric Vella <eric.vella@kla-tencor.com> wrote:

> Here's a more ambitious request, which I don't expect RSI to
> consider, but what the heck: Rewrite IDL as a Perl module. The IDL
> syntax is really annoying, with all sorts of hard to remember quirks and
> strange behavior everywhere.

Well, someone who is locked in Perl-mode is bound to think this way. The converse is also true- a trained IDL user will find Perl to be extremely quirky and counterintuitive. Both languages represent distinctly different approaches, and are not necessarily equally well suited to all tasks.

> This was not so bad when IDL was young and
> had just a few simple commands, but in recent years RSI has expanded IDL
> into a full-blown language. This is a classic example of reinventing
> the wheel, badly. I have no interest in learning a complete new computer
> language, so I will never be an advanced IDL user -- I will just use

When it comes to things like parsing strings, Perl is pretty good. When it comes to any kind of analysis involving arrays, IDL is superior -both from

the standpoint of syntax simplicity, and computational efficiency. The Perl Data Language is an attempt to add IDL functionality to Perl, and while it is useable, it is a far more blatant case of 'reinventing the wheel, badly'.

Having learned C and IDL first, I am probably biased against Perl, but I suspect if you took an average student who had never programmed before, and asked them to write a quick routine to do some data visualization in PDL and IDL, they would have a far easier time with IDL.

If you don't want to learn any new syntax different from Perl, you will probably have a hard time finding any other language to use.

Brad Gom

Sent via Deja.com <http://www.deja.com/>
Before you buy.

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Michael W Asten (masten@mail.earth.monash.edu.au) writes:

> 1) Try harder to keep backward compatibility when releasing new
> versions, eg how silly when the TODAY function disappeared on transition
> from idl5.2 to 5.3 (problem with y2k compliance apparently).

I'm sure it is obvious to anyone who has ever written any code at all, but I'll point it out nonetheless: it is often the case (more often than we would like to admit, probably) that "new features" and "backward compatibility" are mutually exclusive software properties. Certainly anyone perusing these Top 10 Requests has to be struck by how hard it would be to accomplish all that is desired and still be able to run legacy code.

For software that is nearly 20 years old, IDL does a remarkable job of backwards compatibility. True, it is often at the expense of awkwardness and funny syntax. But I for one applaud their efforts.

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Michael W Asten (masten@mail.earth.monash.edu.au) writes:

>
> PS Many thanks David for taking this on. I will send in my \$0.02,
> especially on the need for an upgraded widget generator including
> compound widgets, which I'm sure you will support with impeccably
> unbiased objectivity.

I never claimed unbiased objectivity. All I ever said is
I try to tell the truth as I see it.

Cheers,

David

P.S. But I'll add it to the list...somewhere. :-)

--

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

William (wmc@bas.ac.uk) writes:

> You mean "be put into arrays together", not concatenated, don't you?

Somehow, someone got into the notion of calling this "array concatenation",
but yes, this is what I mean. (I think.)

> If so, could this be extended to allow *any* structures to be put into
> arrays? Is there any particular reason why all array elements have to

> be the same type of structure?

This is the idea behind the changes in IDL 5.4. As long as the array elements "match" in the sense of having the same amount of storage allocated to them, structures can be put into arrays, etc.

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

Vinay L. Kashyap (kashyap@head-cfa.harvard.edu) writes:

> 1. CALL_EXTERNAL

>

> No one I know has come close to implementing a C or Fortran

> call_external in Solaris 2.5+ -- perhaps it's simply a matter

> of providing better documentation.

IDL 5.4 will make this a bit easier by writing the "glue" code between IDL and your C program, as I understand it. This will make the whole argc, argv thing easier to digest.

> 2. _EXTRA

>

> Please consider having all built-in commands accept _EXTRA as a keyword.

Uh, this is the way it works. :-)

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Larry Ashim (kashim@earthlink.net) writes:

- > IDL should have a true WYSIWYG for graphics. Something that allows me to
- > click on an axis displayed on the screen and change the range, font size,
- > etc. Then let me click on the graphic, copy it, and paste it into another
- > application. This would make the preparation of presentations and reports
- > with IDL graphics 100% easier.

IDL Insight. :-)

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Richard Tyc (richt@sbrc.umanitoba.ca) writes:

- > My two bits would be increased performance for Object graphics especially
- > volume objects which do not make use of any OpenGL hardware acceleration so
- > volume rendering speeds are quite slow even on multi CPU machines.
- > This would greatly help in developing medical imaging apps where volume
- > rendering MRI/CT data sets is becoming quite common.

My understanding is that not only are object graphic volumes written to take advantage of OpenGL hardware acceleration, but they are also the only thing in IDL that is multi-tasking. Volume rendering can be parsed out to multiple CPUs on machines that support them.

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

richard hilton (rdh5@dmu.ac.uk) writes:

- > How about an equivalent of the C-commands CONTINUE and BREAK to control
- > loops. This would be more elegant than the "GOTO....label" that exists at
- > the moment. Maybe I'm being pedantic but this seems an obvious change.

Done in IDL 5.4. :-)

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

Theo Brauers (th.brauers@fz-juelich.de) writes:

- > And I am still hoping for a way to get a executable out of my
- > IDL code after compiling without the trouble of setting up a
- > licensed runtime version for a demo or a student lab.

Uh, huh. You and everyone else. But if my take on the situation is anywhere close to accurate, we can all dream on. :-)

Cheers,

David

P.S. RSI has made a start by selling a "development" license. But it is way too expensive and way too restrictive, it seems to me.

--

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Joseph B. Gurman (gurman@gsfc.nasa.gov) writes:

> 3. with NO object interfaces, since it's scientists who are doing a lot
> of the programming, and I can't see why we're paying for features we
> never use (however gnarly).

Uh, Joe. Don't look now, but that kid walking behind you has his eye on *your* job. :-)

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 IDL Requests
Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

Nicolas Decoster (Nicolas.Decoster@Noveltis.fr) writes:

- > * On line object/named-structures development. With IDL 5.3 you can't
- > change a named structure on the fly.

Nor would you want to change a named structure on the fly, IMHO.
Nothing but trouble could possibly result. :-)

- > When you work on a new object
- > graphic or any object or structure, if you add or remove a field you
- > must exit/re-enter IDL for the modification to be effective. It is a bit
- > painfull, and you loose the interest of the use of a scripting language.

Not so. This is the purpose of the .Reset_Session executive
command introduced in IDL 5.3.

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: LINUX Installation Re: Top 10 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

Andrew (noymer@my-deja.com) writes:

- > And since we all know
- > how much David loves *nix, I'm sure it will get top billing ;-)

I've decided to bite the bullet and install some flavor
of LINUX on an old portable I have around here that has
no hope of running Windows 2000.

Any suggestions for the neophyte LINUX installer?

I'm putting it on a DELL Latitude portable.

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

William (wmc@bas.ac.uk) writes:

- > One of my requests would be to throw away the current help system
- > and provide an html-based version.

I may be mistaken, but I think the HTML version of the help system is on the CD-ROM. I remember there was some controversy about whether it should be "installed" with an IDL distribution, but I think RSI still makes it available.

I'll check later. Too much to do now. :-)

- > The other big request would be to make "contour" work properly
- > so that /cell_fill is unnecessary. I mean, in this day and age...

I'm going to confine our "official" requests to those things that have a chance of happening in our lifetimes. :-)

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

Liam E.Gumley (Liam.Gumley@ssec.wisc.edu) writes:

- > I'd settle for encapsulated PostScript output with a decent preview
- > image that could be imported into Word.

I hear--although I haven't seen my IDL 5.4 beta yet--that the Preview image stuff has been "improved". I confess I have heard this before and I am biting my tongue not to say something cynical. But I am hopeful...as always.

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 IDL Requests

Posted by [davidf](#) on Tue, 25 Jul 2000 07:00:00 GMT

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

Oh dear. I had *NO* idea what I was getting myself into!!!!

Anyway, I didn't have newsgroup access last week and now I'm back trying to get caught up with my responsibilities. I can't imagine what I was thinking when I agreed to take this Top Ten project on. :-(

Carsten writes:

- > My absolute top request:
- >
- > Make the plot and axis routines handle DOUBLE values correctly.
- > I am an astronomer, and one out of two numbers I want to plot is
- > larger than 1.e37.

This one, thank goodness, is done. Look for it in IDL 5.4. :-)

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 IDL Requests
Posted by [Michael Asten](#) on Tue, 25 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Not sure what you mean here - do you mean resize a common block during execution of a program, or re-size a common block without exiting the idl system. If it is the latter, then your wish is granted. IDL5.3 has the keyboard command `.reset_session` or `.full_reset_session` which kills a common block (and lots else besides - see help files) and allows you to extend/change, followed by compile, without exit/restart

(Now if only those other 9 items on everybodys wish list were so easy!)

Regards,
Michael Asten

Doug Reynolds wrote:

> I would find it extremely helpful if it were possible to extend COMMON blocks,
> without being forced to exit and restart IDL.
>
> Doug

Subject: Re: Top 10 IDL Requests
Posted by [Joseph B. Gurman](#) on Wed, 26 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <MPG.13e75cd8534ddf9a989b63@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:

> Joseph B. Gurman (gurman@gsfc.nasa.gov) writes:
>
>> 3. with NO object interfaces, since it's scientists who are doing a lot
>> of the programming, and I can't see why we're paying for features we
>> never use (however gnarly).
>
> Uh, Joe. Don't look now, but that kid walking behind
> you has his eye on *your* job. :-)
>

> Cheers,
>
> David

David -

No, that kid can get a `_real_` job as a programmer for a dot.com, and doesn't have to work the same hours for beans as a civil servant scientist. Since programmers are now worth more to society than scientists (as measured purely by salaries), it's clearly a waste of a high-demand resource to have real programmers write code for scientific research. ;-)

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.

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 IDL Requests
Posted by [davidf](#) on Wed, 26 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Doug Reynolds (dsreyn@ll.mit.edu) writes:

> I meant the latter - resizing a common block without exiting IDL. During
> program development, I often find that I want to add a variable to a common
> block. Of course, when I attempt to recompile, IDL says something like this:
>
> % Attempt to extend common block: TEST
>
> It's not particularly difficult to exit and restart IDL, but any variables
> I had defined then need to be reset. Unfortunately, `.reset_session` has the
> same problem, because in addition to common blocks, it appears to kill any
> variables defined at the main level.
>
> I don't know how common blocks are implemented internally (an array of
> pointers, perhaps?), but it seems that if IDL can dynamically redimension
> arrays, it also ought to be able to dynamically resize common blocks also.

It is just that resizing COMMON blocks during the IDL session is such a BAD idea (spend just two seconds thinking of what havoc might ensue) that RSI (to their great credit) refuses to do it, even if they could. Bravo, I say.

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 IDL Requests
Posted by [dsreyn](#) on Wed, 26 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I meant the latter - resizing a common block without exiting IDL. During program development, I often find that I want to add a variable to a common block. Of course, when I attempt to recompile, IDL says something like this:

```
% Attempt to extend common block: TEST
```

It's not particularly difficult to exit and restart IDL, but any variables I had defined then need to be reset. Unfortunately, `.reset_session` has the same problem, because in addition to common blocks, it appears to kill any variables defined at the main level.

I don't know how common blocks are implemented internally (an array of pointers, perhaps?), but it seems that if IDL can dynamically redimension arrays, it also ought to be able to dynamically resize common blocks also.

Doug

In article <397D32B3.6C127E2@mail.earth.monash.edu.au>,
Michael W Asten <masten@mail.earth.monash.edu.au> writes:

```
> Not sure what you mean here - do you mean resize a common block during
> execution of a program, or re-size a common block without exiting the idl
> system. If it is the latter, then your wish is granted. IDL5.3 has the
> keyboard command .reset_session or .full_reset_session which kills a
> common block (and lots else besides - see help files) and allows you
> to extend/change, followed by compile, without exit/restart
```

>
> (Now if only those other 9 items on everybodys wish list were so easy!)
>
> Regards,
> Michael Asten
>
> Doug Reynolds wrote:
>
>> I would find it extremely helpful if it were possible to extend COMMON blocks,
>> without being forced to exit and restart IDL.
>>
>> Doug
>

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Wed, 26 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Larry Ashim (kashim@earthlink.net) writes:

> I cannot believe you suggested IDL Insight as an alternative to a good
> WYSIWYG. I used it ONCE and found it slow, unintuitive and bad. The printed
> graphics looked like something I produced on my Commodore 64 nearly 20 years
> ago. Everyone I talked to at work used Insight only ONCE and had the same
> experience I did. I cannot believe RSI would release Insight in its current
> state.

Oh, Larry. It's just so hard to convey irony in a newsgroup. :-)

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 IDL Requests
Posted by [Larry Ashim](#) on Wed, 26 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I cannot believe you suggested IDL Insight as an alternative to a good

WYSIWYG. I used it ONCE and found it slow, unintuitive and bad. The printed graphics looked like something I produced on my Commodore 64 nearly 20 years ago. Everyone I talked to at work used Insight only ONCE and had the same experience I did. I cannot believe RSI would release Insight in its current state.

As you can tell I am rather opinionated about this, no doubt from 7 years of IDL graphics frustration.

Larry

David Fanning <davidf@dfanning.com> wrote in message
news:MPG.13e75ffbd5a2b51989b67@news.frii.com...

> Larry Ashim (kashim@earthlink.net) writes:

>

>> IDL should have a true WYSIWYG for graphics. Something that allows me to
>> click on an axis displayed on the screen and change the range, font
size,

>> etc. Then let me click on the graphic, copy it, and paste it into
another

>> application. This would make the preparation of presentations and
reports

>> with IDL graphics 100% easier.

>

> IDL Insight. :-)

>

> 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 IDL Requests

Posted by [Nicolas Decoster](#) on Thu, 27 Jul 2000 07:00:00 GMT

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

"Joseph B. Gurman" wrote:

>

> No, that kid can get a _real_ job as a programmer for a dot.com, and
> doesn't have to work the same hours for beans as a civil servant
> scientist. Since programmers are now worth more to society than
> scientists (as measured purely by salaries), it's clearly a waste of a

> high-demand resource to have real programmers write code for scientific
> research. ;-)
>
> 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.

I use objects. I am new in IDL (few month): I was not stuck to old stuff. I use object graphics to draw my figures since the beginning.

In fact I am a "scientist programmer": I don't work for a dot.com, I write code for (my) scientific research. As a programmer I think objects are very usefull to build clean application or data processing. As a programmer I think that some IDL features are badly conceived. As a scientist I think that there are very interesting features, that do all the work without pain. As a scientist sometime I need to know how IDL do the work: I need to inspect the IDL-hidden-code.

Joseph B. Gurman (gurman@gsfc.nasa.gov) writes:

>
> 3. with NO object interfaces, since it's scientists who are doing a lot
> of the programming, and I can't see why we're paying for features we
> never use (however gnarly).

I agree, I don't want to pay for direct graphics: I never use them. :-)

Later.

Nicolas.

--

Tél : 00 (33) 5 62 88 11 16
Fax : 00 (33) 5 62 88 11 12
Nicolas.Decoster@Noveltis.fr

Noveltis
Parc Technologique du Canal
2, avenue de l'Europe
31520 Ramonville Saint Agne - France

Subject: Re: Top 10 IDL Requests
Posted by [Kristian Kjaer](#) on Fri, 28 Jul 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Michael W Asten wrote:
> add a hardcopy print button

Hear!

A print button (and an equivalent cmd-line command) which would work on (direct) graphics already rendered to the screen (using std. direct graphics commands) would null _the_ major quirk in IDL, IMHO.

- Kristian

Kristian Kjær, Risø Natl. Laboratory, Denmark

Subject: Re: Top 10 IDL Requests

Posted by [Craig Markwardt](#) on Sun, 30 Jul 2000 07:00:00 GMT

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

"Mark Hadfield" <m.hadfield@niwa.cri.nz> writes:

>> "Kristian Kjaer" <Kristian.Kjaer@Risoe.Dk> wrote in message

> news:3981DA40.F3BC8FC9@Risoe.Dk...

>>

>> A print button (and an equivalent cmd-line command) which would work

>> on (direct) graphics already rendered to the screen (using std. direct

>> graphics commands) would null _the_ major quirk in IDL, IMHO.

>

> And how would it be done?

>

> Once a direct graphics command has sent output to an output device, the only

> "memory" IDL has of that command is the changed state of the output device.

> At that point the system (or the user) has two ways of recreating the output

> to a different device:

>

> 1. Switch devices & re-issue the same commands

> 2. Read the output back off the device and send it to the new device.

> ...

3. Have the direct graphics window itself store the required data to reproduce the output, and the ability to redirect to a new device.

And I am totally serious; this is what I hacked up with XFWINDOW, which puts a "print" button on any direct graphics window under Unix.

It's a hack because IDL doesn't provide enough documented functionality to achieve the full effect. I had to go stealth. :-)

The professional astronomy package ESO/MIDAS and the plotting program QDP have similar functionality: plot windows can remember their input data, and with a simple command can be redirected to the printer.

The complaint could be made that such a feature might require too much memory, in the case of complex or repeatedly redrawn graphics. There

are pretty simple ways to get around this too. Sigh...

Craig

--

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

Subject: Re: Top 10 IDL Requests

Posted by [Paul van Delst](#) on Mon, 31 Jul 2000 07:00:00 GMT

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

Mark Hadfield wrote:

>
>> "Kristian Kjaer" <Kristian.Kjaer@Risoe.Dk> wrote in message
> news:3981DA40.F3BC8FC9@Risoe.Dk...
>>
>> A print button (and an equivalent cmd-line command) which would work
>> on (direct) graphics already rendered to the screen (using std. direct
>> graphics commands) would null _the_ major quirk in IDL, IMHO.
>
> And how would it be done?

Who cares? This is an implementation (i.e. an RSI) issue.

I just want to make print outs of what I plot by clicking a button.
That's why I smacked together WPLOT and WOPLOT - it has a PS output
button which is enough for me.

See

ftp://airs2.ssec.wisc.edu/pub/paulv/idl/Graphics/direct_graphics/wplot.pro

and

ftp://airs2.ssec.wisc.edu/pub/paulv/idl/Graphics/direct_graphics/woplot.pro

I'm sure there are other codes out there that do the same thing and are
much more sophisticated but this was all I needed. It can't handle log
plots well but I can live with it.

BTW, I'm not recommending anyone grab the above codes for any other
reason than to see how *not* to do something. :o)

paulv

--

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 IDL Requests

Posted by [Mark Hadfield](#) on Mon, 31 Jul 2000 07:00:00 GMT

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

> "Kristian Kjaer" <Kristian.Kjaer@Risoe.Dk> wrote in message
news:3981DA40.F3BC8FC9@Risoe.Dk...

>

> A print button (and an equivalent cmd-line command) which would work
> on (direct) graphics already rendered to the screen (using std. direct
> graphics commands) would null _the_ major quirk in IDL, IMHO.

And how would it be done?

Once a direct graphics command has sent output to an output device, the only
"memory" IDL has of that command is the changed state of the output device.

At that point the system (or the user) has two ways of recreating the output
to a different device:

1. Switch devices & re-issue the same commands
2. Read the output back off the device and send it to the new device.

Number 1 is the method that users normally employ. It's hard to automate
because the graphics commands may well have change the state of the system
in all sorts of unknown ways. This is the method employed by David Fanning's
direct graphics objects. It's a brilliant idea but it has a major
limitation: it requires the user to bundle the graphics command in a single
routine that is "well-behaved", i.e. it can be run repeatedly & is
responsible for recreating all its data every time it is called.

Number 2 is already available for graphics windows via TVRD or the "copy to
clipboard" functionality on Windows. The graphics output is read back from
the output device as a raster image. This is pretty much useless when
redirected to a device with a different resolution.

The "directness" of IDL direct graphics is one of the main reasons for its
speed but also the source of fundamental limitations. It's because of these
fundamental limitations that RSI invented object graphics.

Mark Hadfield
m.hadfield@niwa.cri.nz <http://katipo.niwa.cri.nz/~hadfield/>

Subject: Re: Top 10 IDL Requests
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:

- > I don't know why they haven't grabbed some of
- > the user community's software (or the ideas/needs that drove their
- > generation) and incorporated it in the basic IDL package
- > (licensing/copyright issues notwithstanding).

No comment on this, except to say that I'm sure my
TVIMAGE program, or Liam's IMDISP program, is worth
at least a couple of grand. :-)

- > Maybe we need a more direct line for the IDL user community to the ears
- > of RSI folks that make the decisions than via the chance opportunity
- > that David encountered by meeting an RSI VP at a picnic (or whatever).

I've been sending them an invitation to the annual IDL EPA meeting
for years, but they never show up. I even offered to tone down
the "entertainment" one year, but they wouldn't bite. I think
they should offer to host the event once. I wouldn't be
surprised if they got a whole lot of ideas about things they
might want to work on. :-)

Cheers,

David

P.S. I suggest Beaver Creek in February as a possible venue.
That would really be putting our maintenance dollars to good
use!

--

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 IDL Requests

Mark Hadfield wrote:

```
>
> "Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
> news:3985E8F2.76182471@ncep.noaa.gov...
>> Mark Hadfield wrote:
>>>
>>>> "Kristian Kjaer" <Kristian.Kjaer@Risoe.Dk> wrote in message
>>> news:3981DA40.F3BC8FC9@Risoe.Dk...
>>>>
>>>> A print button (and an equivalent cmd-line command) which would work
>>>> on (direct) graphics already rendered to the screen (using std. direct
>>>> graphics commands) would null _the_ major quirk in IDL, IMHO.
>>>
>>> And how would it be done?
>>
>> Who cares? This is an implementation (i.e. an RSI) issue.
>
> But your solution, like others, doesn't work "on (direct) graphics already
> rendered to the screen ", because the graphics already rendered to the
> screen are now just pixels.
```

I'm not disputing anything you said - I agree with you all the way (hence my attempt and WPLOT result), but at the end of the day I want to be able to say

PLOT, x, y

or

SURFACE, x, y, z

or

MAP_SET, /ISO
MAP_CONTINENTS

and have the graphics window pop up as usual but with a toolbar with "Output to Printer" or "Output to PostScript" buttons. I click on either and the appropriate action is taken.

How that functionality is implemented, e.g. objects graphics, direct graphics saving commands, or direct graphics reading pixels from the screen I absolutely do not care. I just want printable/importable plots. Your objections as to how any of this can be done with the current incarnation of direct graphics commands are noted, but since *I* don't want to write the software that does it, it's a moot point to me (and

most other users I imagine).

The thing I like most about IDL is that it is easy to get stuff on screen to look at it, visualise the data. I don't think it's too much to ask nowadays for a little bit extra extending that to some form of hardcopy(to printer) or softcopy(to PS file) output.

Since I'm harping on about this, I think a graphics window with a mouse driven zoom in/out rubberband box capability (ala DF's xwindow or whatever it's called nowadays) should be *the default* graphics window. Farting about with X/YRANGES on the command line seems terribly clunky.

I think this sort of stuff is something that wouldn't be terribly hard to do (at least by the RSI people) but that would add considerably to the ease of use of IDL. I don't know why they haven't grabbed some of the user community's software (or the ideas/needs that drove their generation) and incorporated it in the basic IDL package (licensing/copyright issues notwithstanding).

Maybe we need a more direct line for the IDL user community to the ears of RSI folks that make the decisions than via the chance opportunity that David encountered by meeting an RSI VP at a picnic (or whatever).

Phew! I'm outta here.

paulv

--

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 IDL Requests

Posted by [Craig Markwardt](#) on Tue, 01 Aug 2000 07:00:00 GMT

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

"Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:

>
> Well now is the time to ask for this feature in WHERE (while IDL 5.4 is
> still in Beta). Perhaps a COMPLEMENT keyword could be added: I'm not
> that fond of optional arguments.
>
> I suggest that anybody who wants this feature send email to RSI
> (mailto:support@rsinc.com) with the subject 'Feature Request' asking for
> this functionality to be added to WHERE.

Umm, where have you *been* the last two weeks? We've been discussing

our favorite wish list items, which David then volunteered to forward to the droids at RSI. David might be overwhelmed by now, but the point of the discussion was indeed to provide feedback to the IDL people about what users want.

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [Liam E. Gumley](#) on Tue, 01 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt wrote:

```
>
> "Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:
>> Michael Plonski wrote:
>>> 1) Complement of Where - so the same call returns where, and a named
>>> variable returns where-not
>>
>> FUNCTION WHERENOT, TEST, COUNT
>> return, where(test eq 0, count)
>> END
> ...
>> It requires another 'where' call if you want the where and where-not,
>> but I'm guessing this functionality isn't needed that often.
>
> No, Mr. Plonski, myself, and several others on this news group were
> asking for the WHERE and its complement in the same function call. It
> does come up a fair amount for me, and it can be a big computation
> expense to do WHERE twice if the arrays are large.
```

Well now is the time to ask for this feature in WHERE (while IDL 5.4 is still in Beta). Perhaps a COMPLEMENT keyword could be added: I'm not that fond of optional arguments.

I suggest that anybody who wants this feature send email to RSI (mailto:support@rsinc.com) with the subject 'Feature Request' asking for this functionality to be added to WHERE.

Cheers,
Liam.

Subject: Re: Top 10 IDL Requests

Posted by [Craig Markwardt](#) on Tue, 01 Aug 2000 07:00:00 GMT

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

"Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:

> Michael Plonski wrote:

>> 1) Complement of Where - so the same call returns where, and a named

>> variable returns where-not

>

> FUNCTION WHERENOT, TEST, COUNT

> return, where(test eq 0, count)

> END

...

> It requires another 'where' call if you want the where and where-not,

> but I'm guessing this functionality isn't needed that often.

No, Mr. Plonski, myself, and several others on this news group were asking for the WHERE and its complement in the same function call. It does come up a fair amount for me, and it can be a big computation expense to do WHERE twice if the arrays are large.

Craig

--

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

Subject: Re: Top 10 IDL Requests

Posted by [Liam E. Gumley](#) on Tue, 01 Aug 2000 07:00:00 GMT

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

Michael Plonski wrote:

> 1) Complement of Where - so the same call returns where, and a named

> variable returns where-not

FUNCTION WHERENOT, TEST, COUNT

return, where(test eq 0, count)

END

b = where(a lt 5, count)

print, b, count

```
0      1      2      3      4
5
```

```
b = wherenot(a lt 5, count)
```

```
print, b, count
```

```
5      6      7      8      9
5
```

It requires another 'where' call if you want the where and where-not, but I'm guessing this functionality isn't needed that often.

Cheers,

Liam.

<http://cimss.ssec.wisc.edu/~gumley>

Subject: Re: Top 10 IDL Requests

Posted by [Michael Plonski](#) on Tue, 01 Aug 2000 07:00:00 GMT

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

1) Complement of Where - so the same call returns where, and a named variable returns where-not

2) Object graphics improvements for 2D - it seems like the only way to get any performance on 2D object graphics is to use the IDL emulation of Open-G.L. The hardware based Open-G.L seems to make you pay all the 3D overhead even if you are only working in a 2D perspective.

3) Better Mapping - specifically a map object that works with object graphics (could be one of those 2D objects from above) and User Defined Mapping. PV-Wave long ago had a user defined map feature though it was not documented to the point that I could ever get it to work. What I would like is to be able to set up a map projection (mostly for raw satellite data) where I pass a function that converts lat/lon to X/Y and vice versa and then all the other map features are available to me. Also Map_set never quite did what you expected. If I recall it used to add something like .01 to the boundary values you gave it. Screwy things happened if you thought you defined a rectangular region in a particular project, but map set didn't (you might get a little 1 or 2 pixel wide triangular wedge along a side that did not have a valid coord_transformation). I found, I always had to read the 4 corner points after using map_set to see what it used since it rarely seemed to use what you asked for. These problems only show up when you are trying to resolve precise coordinates down to the pixel level. A good description of how mapping works and what all those system map variables actually contain also would be nice.

4) Multiple inheritance from objects with a common data field name. I

like that you can explicitly reference an inherited objects methods, why can't they make it so that you can also reference an inherited objects data structure. IDL could follow the same multiple inheritance rules that it applies to methods. I actually don't even care about being able to specifically access all the fields in the inherited objects as unique fields. My primary beef is that the inheritance fails if you have conflicting field names in two different objects. I would be content if they allowed a flag on the inheritance that would just take the field name and data type from first inherited object with that field name and used it. This is what happens when you multiply inherit methods with the same name, except that IDL also allow you to access any of the unique methods if you fully qualify the name with the inherited class. I understand that this would probably be a big performance hit to add this capability to object structures, but it would add some flexibility. I assume that multiple object inheritance just uses relaxed structure concatenation which is probably the cause of this limitation. While we are at it, why not fix this problem on structure concatenation so that you can concatenate structure with conflicting field names, where the first structure with that name get to define the field data type.

5) public, private operations for object inheritance

6) Function Autodefinition files for structures: I hate that you can't set values for structures in autodefinition files. Objects fix this problem somewhat, but then the object structure is private data so you can't access it easily (yes there are workarounds). If RSI does item 5, that would take care of this item. The other alternative is to allow functions for structure autodefinition. If a procedure does the autodefinition you get back a null structure. If a function does the autodefinition you get the named structure with the default values set as returned by the function. It just make the semantics easier than having to call a function to autodefine and set the values for a structure.

7) Integrated support for overlay bit planes. It would be nice to have an easy way to overlay a few bit planes of different colors on top of an image. This is easy to do if you want to give up some of the color values and embed it in the image, but I want a way to toggle - on /off near instantly (like when you load a new 256 bit color map). The current approach used is to embed the overlay into copies of the data into various pixmaps and then load the respective pixmap (ala the old flick routine approach). The problem is I have dozens of bit planes and large images (10s of Megapixels) so you waste a lot of time and memory making pixmaps that may never be displayed. I think the only solution may be the color map for the alpha channel in object graphics, but it would be nice if there was some efficient way to do this in direct graphics.

Maybe someone can explain what a 32-bit true color direct graphics window is, since the channel variable on TV only lets you get to 24 bits (RGB). If we can have 32 bit object graphics, why can we have 32 bit direct graphics with an alpha channel and let IDL deal with how the alpha channel is implemented in an efficient manner. Part of my problem may be that this application is currently using direct graphics with scroll bars for large images, which only a small portion (say 1 Megapixel) is displayed at a time. If I wasn't so lazy I could keep track of what portion is currently displayed and then only update the bit plane overlay for the displayed portion of the data to increase the flicker rate. But then, I would have to keep track of every time the user scrolled and update manually.

8) A class browser for the IDE. It would be nice to be able to see class structures in the IDE. It would be really nice to be able to generate a UML diagram from a project file. If IDL was to publish the description of the project file, it might be able to write some script to extract the classes from the files.

Mike Plonski

Subject: Re: Top 10 IDL Requests
Posted by [Craig Markwardt](#) on Tue, 01 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Paul van Delst <pvandelst@ncep.noaa.gov> writes:

>
> I'm not disputing anything you said - I agree with you all the way
> (hence my attempt and WPLOT result), but at the end of the day I want to
> be able to say
>
> PLOT, x, y
...
>
> and have the graphics window pop up as usual but with a toolbar with
> "Output to Printer" or "Output to PostScript" buttons. I click on either
> and the appropriate action is taken.

I agree. There needs to be a simple way to get from an on-screen rendering to a printer rendering without retyping all the commands, and with reasonable print quality.

My XFWINDOW program is/was a hack, but it shows that what is really

needed in IDL is a way to capture the graphics commands internally.
Then a user-level program can be used to re-render it to the printer.

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [Mark Hadfield](#) on Tue, 01 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
news:398603C8.73ADA35A@ncep.noaa.gov...

> How that functionality is implemented, e.g. objects graphics, direct
> graphics saving commands, or direct graphics reading pixels from the
> screen I absolutely do not care. I just want printable/importable plots.
> Your objections as to how any of this can be done with the current
> incarnation of direct graphics commands are noted, but since *I* don't
> want to write the software that does it, it's a moot point to me (and
> most other users I imagine).

Then you and I agree 100%.

Except that, being a bit of a tinkerer, I would consider writing the
software. And, to some extent I already have*, in object graphics.

(*As, it should be noted have others, notably DF.)

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 IDL Requests
Posted by [Mark Hadfield](#) on Tue, 01 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
news:3985E8F2.76182471@ncep.noaa.gov...

> Mark Hadfield wrote:

>>
>>> "Kristian Kjaer" <Kristian.Kjaer@Riso.Dk> wrote in message
>> news:3981DA40.F3BC8FC9@Riso.Dk...
>>>
>>> A print button (and an equivalent cmd-line command) which would work
>>> on (direct) graphics already rendered to the screen (using std. direct
>>> graphics commands) would null _the_ major quirk in IDL, IMHO.
>>
>> And how would it be done?
>
> Who cares? This is an implementation (i.e. an RSI) issue.

But your solution, like others, doesn't work "on (direct) graphics already rendered to the screen ", because the graphics already rendered to the screen are now just pixels. You can retrieve the pixels; you can (one way or another) save the graphics commands and replay them; for a single purpose application you can hard-wire the command name and just save the data. But you can't get back the shapes you see on the screen--lines, polygons, whatever--because they aren't there any more, they're just pixels.

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 IDL Requests
Posted by [Liam E. Gumley](#) on Wed, 02 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt wrote:
>
> "Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:
>>
>> Well now is the time to ask for this feature in WHERE (while IDL 5.4 is
>> still in Beta). Perhaps a COMPLEMENT keyword could be added: I'm not
>> that fond of optional arguments.
>>
>> I suggest that anybody who wants this feature send email to RSI
>> (<mailto:support@rsinc.com>) with the subject 'Feature Request' asking for
>> this functionality to be added to WHERE.
>
> Umm, where have you *been* the last two weeks? We've been discussing
> our favorite wish list items, which David then volunteered to forward
> to the droids at RSI. David might be overwhelmed by now, but the
> point of the discussion was indeed to provide feedback to the IDL
> people about what users want.

For the last two weeks I've been reading about the dozens of features people want to add to IDL from the simple (adding a complement to WHERE) to the very complex (adding a new command-line friendly object-based graphics system), and the resulting discussions about implementation details. I know RSI people read this newsgroup, but I don't expect them to put features into their flagship product based on our online ramblings (how many posts in the last couple of weeks had 'Top 10' in the subject line?).

Is it reasonable to expect one person (even David) to prioritize all these requests into a Top 10 list? Where do you even start? I suggested that you ask RSI directly because I believe if you send them a very specific request, you have a much better chance of success. If I was running a software company, I'd take a notice of 20 formal email requests for a specific feature (e.g. the WHERE complement).

Cheers,
Liam.
<http://cimss.ssec.wisc.edu/~gumley>

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Wed, 02 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

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

> RSI should be
> (actively) seeking our input. Maybe they do from a select few
> experienced IDL programmers (the board of directors of the Expert
> Programmers Club? :o)

Not likely. Too damn radical, I suspect. :-)

Cheers,

David

P.S. To RSI's credit, I think they listen more now than they used to. But I too am frustrated by lack of attention to core issues. I could do without some of the do-daahs, if the bread an butter stuff (say a TV command that worked like IMDISP) worked properly.

--

David Fanning, Ph.D.
Fanning Software Consulting

Subject: Re: Top 10 IDL Requests
Posted by [Paul van Delst](#) on Wed, 02 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Craig Markwardt wrote:

>
> "Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:
>>
>> Well now is the time to ask for this feature in WHERE (while IDL 5.4 is
>> still in Beta). Perhaps a COMPLEMENT keyword could be added: I'm not
>> that fond of optional arguments.
>>
>> I suggest that anybody who wants this feature send email to RSI
>> (<mailto:support@rsinc.com>) with the subject 'Feature Request' asking for
>> this functionality to be added to WHERE.
>
> Umm, where have you *been* the last two weeks? We've been discussing
> our favorite wish list items, which David then volunteered to forward
> to the droids at RSI. David might be overwhelmed by now, but the
> point of the discussion was indeed to provide feedback to the IDL
> people about what users want.

Last I heard Liam was in Hawaii!! Or Scotland. Or just working at his desk. Which one is it I wonder? :o)

I appreciate David's efforts on the IDL User Community's behalf to forward (some of) the requests for improvements/enhancements/etc of IDL to RSI. At the same time, however, I would like an alternative forum for submitting requests/thoughts to RSI. I wasn't aware that the support email address at RSI could be used for that purpose. I think this "top-ten" list request from users is way overdue - RSI should be (actively) seeking our input. Maybe they do from a select few experienced IDL programmers (the board of directors of the Expert Programmers Club? :o)

In addition to the usual "how to create postscript" posts, I have been reading about a complement to the WHERE function in this newsgroups for years. Where is it? In this respect and others I can understand the attitude of a friend of mine (who wisely spread his eggs in different baskets and hasn't invested a crap-load of time writing IDL code) who recently commented:

"RSI have turned a blind eye to these very obvious

ideas, and it's led me to believe that RSI find the underlying IDL code nearly unmaintainable. You just have to look at their changelogs with each new release to know they aren't tackling the core issues."

It won't make me stop using IDL, but I find it hard to counter his argument when he can do in Python pretty much everything I can in IDL.

Sigh.

paulv

p.s. My friend (who was raised thinking OOP - he's a young 'un) doesn't care too much for the way objects were implemented in IDL either. Damn whippersnapper.

--

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 IDL Requests
Posted by [davidf](#) on Wed, 02 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Michael Plonski (mplonski@aer.com) writes:

> I sent a where-not request to rsinc a few years back and nothing
> happened.

RSI does (believe it or not) respond to user requests.
For example, they added a Min_Value keyword to
the Plot command after I pointed out to them (on several
occasions, I admit) that the *first* thing a new user
tried to do after I introduced the Max_Value keyword was
use a Min_Value keyword.

What RSI does not do well, and this frustrates the hell
out of the customer who is trying to be helpful, is
follow up with a customer to let them know the status
of their request or bug report. I think, frankly, that
this follow-up should be made someone's job. It is
a simple thing to do (you don't have to pay someone
\$100K a year to do it), but it makes S-O-O-O much
difference to a customer. I wouldn't be at all
surprised if it returned \$100K of good will and
increased sales many times over.

And, then, let us KNOW when you fix something or improve something. ADVERTISE it. And do it some place besides the README file that the system administrator deleted to say space for other things. I can't tell you how many IDL programmers I see who go through life writing lousy code because they don't know there is now a better way of doing things. (If I see one more WMENU command I am going to give up and turn my "humorous comment" responsibilities over to Pavel, who has been making me laugh a lot, lately).

Cheers,

David

P.S. Let's just say by making the follow-up someone's job, accountability will have to go up. In fact, if the follow-up were David Stern's job I'll bet a LOT of things would get fixed. :-)

--

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 IDL Requests
Posted by [Michael Plonski](#) on Wed, 02 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

I sent a where-not request to rsinc a few years back and nothing happened. I put together a system that processes Gigabytes of satellite data every day. It has to take several extra passes through the data because there is no way to do efficient nested wheres. Basically, I want to run a new where on the result of the where-not in order to do further segmentation on the items not segmented out by the first where command. I also first asked for user defined map projections probably 6 or 7 years ago. That never happened either, even when I said, well PV-Wave has it (or at least claims to).

If fact the only time I ever got RSI to make a change was when I showed that something in version 5 didn't work the same as version 4 (how the sign bit is promoted in byte to short to long conversions when they introduced the concept of unsigned). It might have been 5.2 .vs. 5.1 instead of 5 .v.s. 4. This seemed to be a high priority for them since

it meant Version 5 wouldn't give the same numerical answer as version 4 in some really limited cases. It did break some code I was using and took a while to track down.

Mike Plonski

"Liam E. Gumley" wrote:

```
>
> Craig Markwardt wrote:
>>
>> "Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:
>>> Michael Plonski wrote:
>>>> 1) Complement of Where - so the same call returns where, and a named
>>>> variable returns where-not
>>>
>>> FUNCTION WHERENOT, TEST, COUNT
>>> return, where(test eq 0, count)
>>> END
>> ...
>>> It requires another 'where' call if you want the where and where-not,
>>> but I'm guessing this functionality isn't needed that often.
>>
>> No, Mr. Plonski, myself, and several others on this news group were
>> asking for the WHERE and its complement in the same function call. It
>> does come up a fair amount for me, and it can be a big computation
>> expense to do WHERE twice if the arrays are large.
>
> Well now is the time to ask for this feature in WHERE (while IDL 5.4 is
> still in Beta). Perhaps a COMPLEMENT keyword could be added: I'm not
> that fond of optional arguments.
>
> I suggest that anybody who wants this feature send email to RSI
> (mailto:support@rsinc.com) with the subject 'Feature Request' asking for
> this functionality to be added to WHERE.
>
> Cheers,
> Liam.
> http://cimss.ssec.wisc.edu/~gumley
```

Subject: Re: Top 10 IDL Requests

Posted by [Paul van Delst](#) on Thu, 03 Aug 2000 07:00:00 GMT

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

Mark Hadfield wrote:

```
>
> "Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
```

> news:398825D1.B3B3F605@ncep.noaa.gov...
>
>> It won't make me stop using IDL, but I find it hard to counter his
>> argument when he can do in Python pretty much everything I can in IDL.
>
> Can he really? Even graphics? Perhaps I'd better take another look at
> Python!

I received another email from him just today (well, overnight - he's pounding away back in Oz). He apparently has nothing useful to do so he decided to replicate (basic) IDL-like functionality in Python. He's also communicating with various other Pythoners to incorporate numerical libraries. He sent me a screen capture and a PS output example of some trig function plots - looks pretty neat. *AND* he has window buttons for output. Only took him 5 days. Pretty flash I reckon. I asked him about images and contour/surface plots etc. I'm interestingly awaiting his reply.

paulv

--

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 IDL Requests
Posted by [Benno Puetz](#) on Thu, 03 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Mark Hadfield wrote:

> "Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
> news:398825D1.B3B3F605@ncep.noaa.gov...
>
>> It won't make me stop using IDL, but I find it hard to counter his
>> argument when he can do in Python pretty much everything I can in IDL.
>
> Can he really? Even graphics? Perhaps I'd better take another look at
> Python!
>

My thoughts - exactly

--

Benno Puetz
Kernspintomographie
Max-Planck-Institut f. Psychiatrie Tel.: +49-89-30622-413

Subject: Re: Top 10 IDL Requests

Posted by [Mark Hadfield](#) on Thu, 03 Aug 2000 07:00:00 GMT

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

"Paul van Delst" <pvandelst@ncep.noaa.gov> wrote in message
news:398825D1.B3B3F605@ncep.noaa.gov...

- > It won't make me stop using IDL, but I find it hard to counter his
- > argument when he can do in Python pretty much everything I can in IDL.

Can he really? Even graphics? Perhaps I'd better take another look at
Python!

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 IDL Requests

Posted by [Jeff Guerber](#) on Thu, 03 Aug 2000 07:00:00 GMT

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

On Fri, 21 Jul 2000, Jeff Guerber wrote:

- > Wow, it looks like we're finally able to post again! And it only
- > took them four months!
- >
- > Anyway, after thinking about it for a while, here's my IDL wishlist.
- > [11 items deleted]

It seems Goddard's about to lose its newsfeed, again, but both
outgoing and incoming this time; so, I'd better get this in while I still
can. I'm afraid I've thought of a couple more wishlist items to heap upon
that ever-growing pile:

12) Something like "extra", but which does not override keywords
explicitly specified on the call line; "default", perhaps. I recently
wrote a program that instantiates an object (which I also wrote), then
adds various things to it (the object uses David Fanning's exceedingly
useful `linkedlist__define.pro`... Thanks David!), each of which has
properties (specified with keywords, of course). The calling program has

a structure containing defaults for the properties, which I pass to the object using "_extra="; however, on some of the calls, I would have liked to have overridden some of these defaults. But since _extra overrides explicit keywords, I wound up making extra calls to the object's SetProperty method.

13) The ability to move up and down the call stack from the command line. When a program stops in a procedure, you can examine the variables in that procedure; but, sometimes I'd like to examine some of the ones in its caller, too. I can do this in IDLDE (I think); but since I normally use emacs+idlwave (on HP-UX), I'd much rather be able to do it from the regular command line. Somewhat like you can in, say, gdb or other debuggers.

OK, so it's only supposed to be a "Top 10" list, but hey, 10 EQ 13 for sufficiently large values of "10"! :-)

Jeff Guerber

Raytheon ITSS
NASA Goddard Space Flight Center
Oceans & Ice Branch (code 971)

Any opinions here are my own. Well, in a few cases I borrowed them from other people, but they're certainly not Raytheon's or NASA's!

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

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

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

> Not by "somebody" but by David Fanning, whos books on IDL are considered
> one of the best sources of information about the language. As far as I
> know, David's expertise in IDL is very high, and even RSI uses his
> services. David's SET* routines are short, fast and efficient, and
> available as easily as IDL distribution.

Well, I'd love to take the credit for these, but
they were actually contributed by a secret source
at RSI. :-)

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: Multiple Inheritance (Top 10 IDL Requests)
Posted by [Martin Schultz](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

One topic that didn't let me off the hook:

Michael Plonski wrote:

>
>
>
> 4) Multiple inheritance from objects with a common data field name. I
> like that you can explicitly reference an inherited objects methods, why
> can't they make it so that you can also reference an inherited objects
> data structure. IDL could follow the same multiple inheritance rules
> that it applies to methods. I actually don't even care about being able
> to specifically access all the fields in the inherited objects as unique
> fields. My primary beef is that the inheritance fails if you have
> conflicting field names in two different objects.

A potential solution to this problem would be aliasing. Fortran90 has a very nice syntax for modules which would accomodate multiple object inheritance easily. How about something like:

```
object = { class_name, $
           INHERITS base_class ONLY=(index, name=>basename), &
           INHERITS other_class ONLY=(name) }
```

Here, "index" would be taken from base_class as is, but base_class' "name" tag would be referenced as "basename" in the new object, and other_class would also provide a "name" field which would be seen as "name". The ONLY specifier would also add some "privacy" to the objects. It could perhaps be supplemented with an ALIAS "keyword" which would not exclude other fields but only do the aliasing .

... and I also second this one:

- > 5) public, private operations for object inheritance

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 IDL Requests
Posted by [Craig Markwardt](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Alex Schuster <alex@pet.mpin-koeln.mpg.de> writes:

```
> 5) The complement to WHERE() is great. Now I would like routines which
> give me the intersection of two arrays, the union, and the difference.
> These already exist (setintersection.pro etc.), but they were written by
```

- > somebody and are not part of the standard distribution.
- > I'd also like something like an IN operator: a IN b gives 1, if the
- > scalar a is in the array b.

I agree this would be nice notation. I have a function CMSET_OP on my web page which is more full-featured and robust than the RSI set functions: it has "union," "intersect," and "exclusive or," plus it works on string and float values. There was a big discussion a few months ago about these techniques, with JD Smith being a big contributor.

In my own personal version of CMSET_OP I have the capability to extract array indices rather than array values, which was a requested feature in the discussion. When the current crush of urgent work subsides, I will post that updated version (a few days).

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [promashkin](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Alex Schuster wrote:

- > Now I would like routines which
- > give me the intersection of two arrays, the union, and the difference.
- > These already exist (setintersection.pro etc.), but they were written by
- > somebody

Not by "somebody" but by David Fanning, whos books on IDL are considered one of the best sources of information about the language. As far as I know, David's expertise in IDL is very high, and even RSI uses his services. David's SET* routines are short, fast and efficient, and available as easily as IDL distribution.

Cheers,
Pavel

Subject: Re: Top 10 IDL Requests

Posted by [Alex Schuster](#) on Fri, 04 Aug 2000 07:00:00 GMT

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

I'll add some suggestions, too.

1) Improve the widgets. I see this barely mentioned in this group, but: they suck. They really do. Why do I have to use hacks to get keyboard events? This is ridiculous. And it doesn't work for unusual keys like Alt-F3. I haven't tried this, but I guess even with menu bars an Alt-F doesn't open the File menu, does it?

And they don't look good. When I create a row base and some widget_sliders, cw_fields and cw_bgroups with labels, then the labels have a different vertical position. At least in IDL 5.2 under Unix which I am still using, but I would be surprised if this was changed in 5.3.

So when I want a really good-looking interface, I have to create some small extra bases in which these widgets have an y offset.

Is the 'widget TLB not resizing properly' bug fixed yet? Well, I bet it isn't. And for me the trick with setting the UPDATE keyword doesn't always seem to work.

2) I also would prefer it IDL would not remove trailing dimensions of 1 in some cases. I quite often use code like

```
img = reform( fltarr( x, y, z ), x, y, z )
```

because z might be 1, but I still want to access the data as 3d image.

Of course old code has to work unchanged, so maybe something like `img=fltarr(x,y,z,/strict)` would do the job.

3) It would be nice if RANDOM() would work correctly in ALL versions of IDL (including IDL x.y.0), not only in IDL x.y.1. Such bugs make me feel sad. I use IDL as a scientific tool, and I need to rely on the correct implementation of the mathematical routines.

4) When I report a bug to RSI, it would be nice if I got a notification that it is fixed.

5) The complement to WHERE() is great. Now I would like routines which give me the intersection of two arrays, the union, and the difference.

These already exist (setintersection.pro etc.), but they were written by somebody and are not part of the standard distribution.

I'd also like something like an IN operator: `a IN b` gives 1, if the scalar a is in the array b.

I think I forgot many points, but the list is long enough already.

Alex

--

Alex Schuster Wonko@weird.cologne.de
alex@pet.mpin-koeln.mpg.de

PGP Key available

Subject: Re: Top 10 IDL Requests
Posted by [Stephen Serjeant](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

Jeff Guerber wrote:

```
> 13) The ability to move up and down the call stack from the
> command line.
```

Craig Markwardt wrote an excellent package to do just this, called dxdebug - see <http://cow.physics.wisc.edu/~craigm/idl/idl.html>
It's based around the undocumented ROUTINE NAMES() function.

Just one of the many incredibly useful things you pick up from lurking in this newsgroup...

steve serjeant

Dr Stephen Serjeant
Astrophysics Dept
Imperial College London
Blackett Laboratory
Prince Consort Road
London SW7 2BW
UK

Tel: +44 20 7594 7534
Mobile: +44 7946 605 913
Fax: +44 20 7594 7541

s.serjeant@ic.ac.uk
or s.serjeant1@physics.oxford.ac.uk
<http://icstar5.ph.ic.ac.uk/~sbqs>

Subject: Re: Top 10 IDL Requests
Posted by [Nicolas Decoster](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Paul van Delst wrote:

>
> I received another email from him just today (well, overnight - he's
> pounding away back in Oz). He apparently has nothing useful to do so he
> decided to replicate (basic) IDL-like functionality in Python. He's also
> communicating with various other Pythoners to incorporate numerical
> libraries. He sent me a screen capture and a PS output example of some
> trig function plots - looks pretty neat. *AND* he has window buttons for
> output. Only took him 5 days. Pretty flash I reckon. I asked him about
> images and contour/surface plots etc. I'm interestingly awaiting his

> reply.

This sounds interesting. I am curious to see such a thing. Can you tell me how I can contact this guy ?

Thanks in advance.

Nicolas.

--

Ti ½l. : 00 (33) 5 62 88 11 16
Fax : 00 (33) 5 62 88 11 12
Nicolas.Decoster@Noveltis.fr

Noveltis
Parc Technologique du Canal
2, avenue de l'Europe
31520 Ramonville Saint Agne - France

Subject: Re: Top 10 IDL Requests
Posted by [Mark Hadfield](#) on Fri, 04 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

> On Fri, 21 Jul 2000, Jeff Guerber wrote:
>
> 12) Something like "_extra=", but which does _not_ override keywords
> explicitly specified on the call line; "_default=", perhaps.

I think the way you want _extra to work is the way it *should* work and I don't think it would break too much code if it *did* work that way. So I don't support "_default".

I raised this with RSI support some time ago and they said they'd look into it. It all depends on how on the order in which IDL handles duplicate tags in structures (at least that's what I remember thinking the last time I thought about it, but that was a long time ago and I may have become confused in my old age).

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 IDL Requests

Posted by [Nick Bower](#) on Fri, 04 Aug 2000 07:00:00 GMT

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

> \But your solution, like others, doesn't work "on (direct) graphics already
> rendered to the screen ", because the graphics already rendered to the
> screen are now just pixels. You can retrieve the pixels; you can (one way or

That is a simplistic solution to plotting. In order to satisfy Paul's need (something I definately would like to see too) graphics would be written to a virtual metafile and not a standard pixmap. So the sensible thing would be that all plot commands drew vectors to the virtual metafile, and then the viewport on each platform simply renders this as a pixmap, translating vector to raster.

If it were done this way, you could implement highly functional zooming and postscript dumps in the viewport which simply go back and access the vector information in the virtual metafile. I've seen matlab do it, and I'd say this highlights the difference in the visualization approaches and considerations the two packages have made. It's clear to me which approach is a more sophisticated.

nick

Subject: Re: Top 10 IDL Requests

Posted by [dominik](#) on Mon, 07 Aug 2000 07:00:00 GMT

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

> On Fri, 21 Jul 2000, Jeff Guerber wrote:

>

> 13) The ability to move up and down the call stack from the command
> line. When a program stops in a procedure, you can examine the
variables

> in that procedure; but, sometimes I'd like to examine some of the ones
in

> its caller, too. I can do this in IDLDE (I think); but since I
normally

> use emacs+idlwave (on HP-UX), I'd much rather be able to do it from
the

> regular command line. Somewhat like you can in, say, gdb or other
> debuggers.

You can do this in IDLWAVE, too. Check the "Debug" menu for "Stack up" and "Stack down", and use the "print expression" command to print the expression on a higher level.

- Carsten

--

Carsten Dominik <dominik@astro.uva.nl> _ /
Sterrenkundig Instituut "Anton Pannekoek" |X|
Kruislaan 403; NL-1098 SJ Amsterdam /| | _ _ _ / \
phone +31 (20) 525-7477; FAX +31 (20) 525-7484 _|o|_ / ~ ~ _ / ~ ~ ~

Sent via Deja.com <http://www.deja.com/>
Before you buy.

Subject: Re: Top 10 IDL Requests
Posted by [Craig Markwardt](#) on Thu, 10 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

ottehoman@my-deja.com writes:

> In article <397df38e.0@news.nwl.ac.uk>,
> wmc@bas.ac.uk wrote:
> ;[other stuff]
>
>> Ah, no, thats not enough. I want, say,
> structures which represent a data
>> agglomeration (I'm trying not to say object)
> with loads of header fields
>> the same, but a few fields (maybe just the one
> "data" field) different.
>> So the storage is different. I know I could do
> this by putting a pointer
>> into the structure instead, but... I can't see
> why IDL shouldn't do this itself.
>> I guess I'm assuming that, when IDL stores an
> array of strucutres, it doesn't
>> store the structures consecutively anyway - just
> pointers to the structures.
>> In which case, it shouldn't matter what the
> structure types are. I think.
>
> William, David,
>
> I have the same problem - I'd like to create a
> hierarchical data structure, basically an array,
> of structures (so I can use DataFile(1),
> DataFile(2), etc...) Each structure consists of
> headers, parameters, and *dynamic* arrays. So the
> structures are *more or less* the same, but
> differ in the lenght of their arrays. Using IDL

- > 5.3 (sorry, our site has only this
- > version licensed) the contents of an array can
- > only be of one single type. My filestructures
- > are like this: {{header},{parameters},{data}},
- > where {data} is a structure with arrays of (from
- > file to file) different lengths. I only know at
- > runtime how long these arrays are.

I hear your problems. I run into the same kinds of things myself. It turns out that all structures must have exactly the same format, including the number of elements in every array. I believe that arrays of structures *are* in fact stored in a single contiguous memory block, except for strings. If you ever notice that manipulations with large structures are very slow, that's why.

To get to the solution, I think David, JD, etc are right. You need to use pointers. If you need compatibility with IDL 4, then you can use handles. You get the benefit of both pointers and handles via Liam Gumley's POINTER_* family of functions.

I would recommend that to start you use as few pointers as you can get away with. Unlike normal IDL variables, there are real penalties for forgetting to free a pointer when you are done. One possibility is to wrap all of your variable sized data from a file into a single anonymous structure, and make a single tag in your fixed-sized structure point to it. As I said, large amounts of data in a structure tend to get very sloooow though.

Craig

--

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

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Thu, 10 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

J.D. Smith (jdsmith@astro.cornell.edu) writes:

- > I have sensed some great hesitation over
- > the use of pointers for complex data structures.

Uh, have you mentioned the word "objects" lately? Uh, oh.
You just did. :-(

> This doesn't even go into the inherited data members.

Please, JD, it is not even close to Halloween.

> The deepest reference is in 2f. and goes something like

>

> `*(self.DR)[i].HEADER)[j]`

>

> which may frighten you, but I assure you after a bit of practice and review of
> the (unwritten) precedence of "!", is quite tractable.

I actually think the precedence thing *is* written down...
somewhere. You just have to be an extremely lucky cultural
anthropologist to discover it. :-)

> Happy pointering.

Indeed.

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 IDL Requests

Posted by [John-David T. Smith](#) on Thu, 10 Aug 2000 07:00:00 GMT

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

David Fanning wrote:

>

> Otte Homan (ottehoman@my-deja.com) writes:

>

>> I have the same problem - I'd like to create a
>> hierarchical data structure, basically an array,
>> of structures (so I can use `DataFile(1)`,
>> `DataFile(2)`, etc...) Each structure consists of
>> headers, parameters, and *dynamic* arrays. So the
>> structures are *more or less* the same, but
>> differ in the length of their arrays. Using IDL
>> 5.3 (sorry, our site has only this

>> version licensed) the contents of an array can
>> only be of one single type. My filestructures
>> are like this: {{header},{parameters},{data}},
>> where {data} is a structure with arrays of (from
>> file to file) different lengths. I only know at
>> runtime how long these arrays are.
>>
>> Any solution ?
>
> Well, I reiterate. Pointers. The solution is pointers. :-)
>
> If the data field of this structure is a pointer to
> the variable length array, then you can store as many
> of these structures in an array as you like.
>
> William asked for an array of different structures,
> which sort of turns the definition of an array
> topsy-turvy. But *this* problem can be solved with
> pointers, I'm sure of it. :-)
>
> Cheers,
>
> David

I have sensed some great hesitation over the use of pointers for complex data structures. To ease the feeling that you'll be lost in a maze of no return, I post here a summary of a single data structure of mine, which, while at first glance unwieldy, is actually quite flexible and reasonably easy to use. To summarize:

INHERIT'ing object class scoreProj containing:

1. various "regular" numerical and string data member fields.
2. pointer to a dynamic list of struct of type SCORE_DR
SCORE_DR containing:
 - a. Various regular fields.
 - b. Pointer to a dynamically sized array of strings (filenames)
 - c. Pointer to a data array of size 128x128xn (n determined at runtime)
 - d. Pointer to a data array of size 128x128x2
 - e. Pointer to dynamically sized list of planes.
 - f. Pointer to dynamic array of pointers to dynamic string arrays.
3. pointer to a struct of type SCORE_STACK
SCORE_STACK containing:
 - a. Various regular fields
 - b. 4 pointers to data arrays of size 128x128x2
 - d. Pointer to dynamic list of floating pairs (2xn)
4. pointer to a struct of type SCORE_EXTRACT
SCORE_EXTRACT containing:

- a. Various regular fields
- b. 3 pointers to data of size 128x128
- c. A pointer to data of size 3xn
- 5. pointer to a struct of type scoreProj_wlInfo
scoreProj_wlInfo containing:
 - a. Various widget id's as longs.
 - b. Pointer to dynamic array of button id's.

This doesn't even go into the inherited data members.

The deepest reference is in 2f. and goes something like

```
*(/*self.DR)[i].HEADER)[j]
```

which may frighten you, but I assure you after a bit of practice and review of the (unwritten) precedence of "/*", is quite tractable.

Happy pointering,

JD

--

J.D. Smith /*\ WORK: (607) 255-6263
Cornell University Dept. of Astronomy */ (607) 255-5842
304 Space Sciences Bldg. /*\ FAX: (607) 255-5875
Ithaca, NY 14853 */

Subject: Re: Top 10 IDL Requests
Posted by [davidf](#) on Thu, 10 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Otte Homan (ottehoman@my-deja.com) writes:

- > I have the same problem - I'd like to create a
- > hierarchical data structure, basically an array,
- > of structures (so I can use DataFile(1),
- > DataFile(2), etc...) Each structure consists of
- > headers, parameters, and *dynamic* arrays. So the
- > structures are *more or less* the same, but
- > differ in the lenght of their arrays. Using IDL
- > 5.3 (sorry, our site has only this
- > version licensed) the contents of an array can
- > only be of one single type. My filestructures
- > are like this: {{header},{parameters},{data}},
- > where {data} is a structure with arrays of (from
- > file to file) different lengths. I only know at
- > runtime how long these arrays are.

>
> Any solution ?

Well, I reiterate. Pointers. The solution is pointers. :-)

If the data field of this structure is a pointer to the variable length array, then you can store as many of these structures in an array as you like.

William asked for an array of different structures, which sort of turns the definition of an array topsy-turvy. But **this** problem can be solved with pointers, I'm sure of it. :-)

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 IDL Requests
Posted by [ottehoman](#) on Thu, 10 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <397df38e.0@news.nwl.ac.uk>,
wmc@bas.ac.uk wrote:
;[other stuff]

> Ah, no, thats not enough. I want, say,
structures which represent a data
> agglomeration (I'm trying not to say object)
with loads of header fields
> the same, but a few fields (maybe just the one
"data" field) different.
> So the storage is different. I know I could do
this by putting a pointer
> into the structure instead, but... I can't see
why IDL shouldn't do this itself.
> I guess I'm assuming that, when IDL stores an
array of strucutres, it doesn't
> store the structures consecutively anyway - just
pointers to the structures.

> In which case, it shouldn't matter what the structure types are. I think.

William, David,

I have the same problem - I'd like to create a hierarchical data structure, basically an array, of structures (so I can use DataFile(1), DataFile(2), etc...) Each structure consists of headers, parameters, and *dynamic* arrays. So the structures are *more or less* the same, but differ in the length of their arrays. Using IDL 5.3 (sorry, our site has only this version licensed) the contents of an array can only be of one single type. My filestructures are like this: {{header},{parameters},{data}}, where {data} is a structure with arrays of (from file to file) different lengths. I only know at runtime how long these arrays are.

Any solution ?

Otte Homan
Swiss Federal Institute of Technology
Zurich, Switzerland

Sent via Deja.com <http://www.deja.com/>
Before you buy.

Subject: Re: Top 10 IDL Requests
Posted by [ottehoman](#) on Fri, 11 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear David,

In article <MPG.13fc6344e757c00b989bb0@news.frii.com>, davidf@dfanning.com (David Fanning) wrote:

> Otte Homan (ottehoman@my-deja.com) writes:
>
>> I have the same problem - I'd like to create a
[...]
>> Any solution ?
>
> Well, I reiterate. Pointers. The solution is pointers. :-)
>
> If the data field of this structure is a pointer to

> the variable length array, then you can store as many
> of these structures in an array as you like.
>
> William asked for an array of different structures,
> which sort of turns the definition of an array
> topsy-turvy. But *this* problem can be solved with
> pointers, I'm sure of it. :-)
>
>

But then why use IDL at all? Why not directly use g++/gtk
or gnuplot or any other OO-language ?

Alright, let's point then.

Thanx for the help.

Otte

Sent via Deja.com <http://www.deja.com/>
Before you buy.

Subject: Bottom 10 IDL requests ;-) was: Re: Top 10 IDL Requests
Posted by [wmc](#) on Fri, 11 Aug 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

ottehoman@my-deja.com wrote:

> In article <MPG.13fc6344e757c00b989bb0@news.frii.com>,
> davidf@dfanning.com (David Fanning) wrote:

>> Well, I reiterate. Pointers. The solution is pointers. :-)

>> William asked for an array of different structures,
>> which sort of turns the definition of an array
>> topsy-turvy.

> But then why use IDL at all? Why not directly use g++/gtk
> or gnuplot or any other OO-language ?

OK, yes, I could use pointers. But its an extra layer of complication.
IDL is nice because a lot of stuff can be done easily. Now, why cannot
there be an array type which contains different types? The array would
really be pointers to the things in it, automatically created and hidden.
Just a wish, I know...

In fact, if IDL are listening, null arrays are higher up the list, I

really want to be able to write

a=d(where(b eq c))

not the 2-line version...

-W.

--

William M Connolley | wmc@bas.ac.uk | <http://www.nerc-bas.ac.uk/icd/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself

Subject: Re: Top 10 IDL Requests

Posted by [davidf](#) on Sat, 12 Aug 2000 07:00:00 GMT

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

Otte (ottehoman@my-deja.com) writes:

> But then why use IDL at all? Why not directly use g++/gtk
> or gnuplot or any other OO-language ?

Certainly. Be my guest. But then you *really* have
to know about pointers. :-)

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
