Subject: Re: Accessing "main" level variables
Posted by Craig Markwardt on Tue, 18 Jan 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Michael Lefsky <lefsky@home.com> writes:

> 2) Is it possible to access main level variables from within a procedure?
> For instance, to solve the problem outlined in 1), you could simply
> write a procedure called "save" which would have precedence over the
> system command save. However, once in the procedure "save", you would no
> longer have access to the main level variables that you want to save.
> Even if you specified them as arguments to "save" you would no longer
> have access to their original (ie. main level) names. There are numerous
> applications for which I would like to be able to somehow get either)
> the names of the variables passed to the current procedure (ie. the
> names they had at the level above them) or to access the main level
> variables. Of course there might be numerous ways to do either of these
> things, from the ugly to the sublime- personally, I am not particular.

I like your questino. I know I will get flamed for saying it, but I
think there are some times when it is desireable for a program to
deposit its results into the main level. For example, a fully canned
curve fitting program may want to return the fitting results. This
should of course be done with full control and permission of the user.

"Why," you ask, "don't you just return it as a regular or keyword
parameter?"

"Simple," I say. "You don't always know how many variables you are
going to return." Also, if it were a curve fitting program "for
dummies" then I would want to make it as simple as possible.

Incidentally, there is an internal IDL routine which *can* do this.
Unfortunately you need to write a DLM to access it.

Craig

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

Subject: Re: Accessing "main" level variables
Posted by mallors on Tue, 18 Jan 2000 08:00:00 GMT
View Forum Message <> Reply to Message

In article <3883C97E.8DA692CE@home.com>,
Michael Lefsky <lefsky@home.com> writes:

> I have two questions, one short and direct, one longer and more diffuse.
>
> 1) Has anyone found a way to keep IDL from either overwriting existing
> idlsave.dat files or not allowing a user to access the save function
> without specifiying a filename? If I had a dollar for every time I had
> to ask my system administrator to restore an old "idlsave.dat" file
> because I had accidently overwritten it, I'd have maybe 4 or 5 bucks.
> Admittdly that's not much, but it is enough to buy my SA a slice of
> pizza, which would go a long way to apologizing for all those
> restores:).
>
> 2) Is it possible to access main level variables from within a procedure?
> For instance, to solve the problem outlined in 1), you could simply
> write a procedure called "save" which would have precedence over the
> system command save. However, once in the procedure "save", you would no
> longer have access to the main level variables that you want to save.
> Even if you specified them as arguments to "save" you would no longer
> have access to their original (ie. main level) names. There are numerous
> applications for which I would like to be able to somehow get either)
> the names of the variables passed to the current procedure (ie. the
> names they had at the level above them) or to access the main level
> variables. Of course there might be numerous ways to do either of these
> things, from the ugly to the sublime- personally, I am not particular.
>
> Does anyone have any thoughts on this?

It's not exactly what you describe, but perhaps you could
make use of this routine:

http://cspar.uah.edu/~mallozzir/software/idl/info/xvarexport .html

Regards,

-bob

--
~~~~~~ ~~~~~~~~~
Robert S. Mallozzi 256-544-0887
Mail Code SD 50
http://gammaray.msfc.nasa.gov/ Marshall Space Flight Center
http://cspar.uah.edu/~mallozzir/ Huntsville, AL 35812
~~~~~~ ~~~~~~~~~

Subject: Re: Accessing "main" level variables

Posted by Michael Lefsky on Wed, 19 Jan 2000 08:00:00 GMT
View Forum Message <> Reply to Message

<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
Craig Markwardt wrote:
<p>>Incidentally, there is an internal IDL routine which *can* do this.
<br>>Unfortunately you need to write a DLM to access it.
<p>I&nbsp;haven't worked with the internal IDL&nbsp;routines, and don't
plan to. But I did look at the EDG documentation,
<br>and it seems like it would be relatively straighforward to write a
toolbox&nbsp; that would allow us (the IDL&nbsp;user community)&nbsp;to
access main level variables from within procedures. For instance, an IDL
shell could
<br>be written to access the routine
<p> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp; &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;&nbsp;
IDL_VPTR IDL_GetVarAddr(char *name)
<p>which would return the address of the variable, and then the variable
could be obtained and returned as the return value of the function.
<p>Has anyone written such a toolbox?
<br>&nbsp;
<p>Craig Markwardt wrote:
<blockquote TYPE=CITE>Michael Lefsky &lt;lefsky@home.com> writes:
<p>> 2) Is it possible to access main level variables from within a procedure?
<br>> For instance, to solve the problem outlined in 1), you could simply
<br>> write a procedure called "save" which would have precedence over
the
<br>> system command save. However, once in the procedure "save", you would
no
<br>> longer have access to the main level variables that you want to save.
<br>> Even if you specified them as arguments to "save" you would no longer
<br>> have access to their original (ie. main level) names. There are numerous
<br>> applications for which I would like to be able to somehow get either
)
<br>> the names of the variables passed to the current procedure (ie. the
<br>> names they had at the level above them) or to access the main level
<br>> variables. Of course there might be numerous ways to do either of
these
<br>> things, from the ugly to the sublime- personally, I am not particular.
<p>I like your questino.&nbsp; I know I will get flamed for saying it,
but I
<br>think there are some times when it is desireable for a program to
<br>deposit its results into the main level.&nbsp; For example, a fully
canned
<br>curve fitting program may want to return the fitting results.&nbsp;
This
<br>should of course be done with full control and permission of the user.

<p>"Why," you ask, "don't you just return it as a regular or keyword
<br>parameter?"
<p>"Simple," I say.&nbsp; "You don't always know how many variables you
are
<br>going to return."&nbsp; Also, if it were a curve fitting program "for
<br>dummies" then I would want to make it as simple as possible.
<p>Incidentally, there is an internal IDL routine which *can* do this.
<br>Unfortunately you need to write a DLM to access it.
<p>Craig
<p>--
<br> -- --------------
<br>Craig B. Markwardt, Ph.D.&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;& nbsp;&nbsp;&nbsp;
EMAIL:&nbsp;&nbsp;&nbsp; craigmnet@cow.physics.wisc.edu
<br>Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
<br> -- -------------- </blockquote>

<pre>--&nbsp;
Michael Lefsky, PhD
Research Ecologist
U.S. Forest Service
Forest Sciences Laboratory
Corvallis, OR&nbsp;
lefsky@fsl.orst.edu
<A HREF="http://www.fsl.orst.edu/~lefsky">http://www.fsl.orst.edu/~lefsky</A></pre>
&nbsp;</html>

Subject: Re: Accessing "main" level variables

Posted by Craig Markwardt on Wed, 19 Jan 2000 08:00:00 GMT

View Forum Message <> Reply to Message

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

> Craig Markwardt wrote:
>>
>> [...]
>>
>> I like your questino. I know I will get flamed for saying it, but I
>> think there are some times when it is desireable for a program to
>> deposit its results into the main level. For example, a fully canned
>> curve fitting program may want to return the fitting results. This
>> should of course be done with full control and permission of the user.
>> ... portion deleted ...
>
> No! No flaming. But as you describe it, the solution to your problem
> would
> sound like almost the typical object oriented approach. Simple enough,
> too,

> once the user understands he/she has to write
>   obj_fit->get_parameter,a0=a0,m=m
> (or something similar)
> ... portion deleted ...

I agree with you from a *programmer's* point of view. A programmer
wants a clean programming interface with predictable behavior.
Clearly a procedure that modifies external variables goes against this
ideal.

HOWEVER, from a lowly *user's* perspective, I assert that sometimes it
would be nice for a program to manipulate some main level variables
for me. I know, because I'm both a programmer and a user. Querying
objects as you suggest, while linguistically pure, will get stale
pretty quickly while I am working interactively at the command line.

In the back of my mind I would like to write a nice graphical curve
fitting application using MPFIT, and I want it to be simple simple
simple. The question is, how do I get the results out once the fit is
finished?

Craig

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

Subject: Re: Accessing "main" level variables
Posted by Craig Markwardt on Wed, 19 Jan 2000 08:00:00 GMT

View Forum Message <> Reply to Message

Carsten Dominik <dominik@astro.uva.nl> writes:

>>>> >> "CM" == Craig Markwardt <craigmnet@cow.physics.wisc.edu> writes:
>
> CM> Incidentally, there is an internal IDL routine which *can* do
> CM> this. Unfortunately you need to write a DLM to access it.
>
> Is this documented somewhere?

Yes, it's in the *External Development Guide*, in the chapter called
Variables. The function name is IDL_GetVarAddr(), but as I said this
is an internal C function that would need to be accessed by a DLM.

Craig

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

Subject: Re: Accessing "main" level variables
Posted by Martin Schultz on Wed, 19 Jan 2000 08:00:00 GMT

View Forum Message <> Reply to Message

Craig Markwardt wrote:
>
>
>
> I like your questino. I know I will get flamed for saying it, but I
> think there are some times when it is desireable for a program to
> deposit its results into the main level. For example, a fully canned
> curve fitting program may want to return the fitting results. This
> should of course be done with full control and permission of the user.
>
> "Why," you ask, "don't you just return it as a regular or keyword
> parameter?"
>
> "Simple," I say. "You don't always know how many variables you are
> going to return." Also, if it were a curve fitting program "for
> dummies" then I would want to make it as simple as possible.
>
> Incidentally, there is an internal IDL routine which *can* do this.
> Unfortunately you need to write a DLM to access it.
>
> Craig
>
> --
> -- --------------
> Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> -- --------------

No! No flaming. But as you describe it, the solution to your problem
would
sound like almost the typical object oriented approach. Simple enough,
too,
once the user understands he/she has to write
   obj_fit->get_parameter,a0=a0,m=m
(or something similar)
The nice thing would be that you decouple the actual fitting process

from the
output, so you can access the fit result at any time (until you
obj_destroy
the fit object, of course). You could still write a
procedural/functional
interface for the object to prevent the "uninitiated" user from running
away.

Cheers,
Martin

PS: maybe I shut put a foot in my mouth, for I am talking so much about
objects these days but never supply any useful routines which make use
of them ;-(One day that will change for sure!...

```
--
 [[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[ [[[[[[[
[[ 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: Accessing "main" level variables
Posted by Carsten Dominik on Wed, 19 Jan 2000 08:00:00 GMT

View Forum Message <> Reply to Message

>>>> > "CM" == Craig Markwardt <craigmnet@cow.physics.wisc.edu> writes:

CM> Incidentally, there is an internal IDL routine which *can* do
CM> this. Unfortunately you need to write a DLM to access it.

Is this documented somewhere?

- 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|____/ ~~ \___/    ~~~~~
```