

---

Subject: Re: global singleton object ?

Posted by [David Fanning](#) on Tue, 02 Oct 2001 15:55:55 GMT

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

---

R.G.S. (rgs1967@hotmail.com) writes:

> partly inspired by the recent threads on global variables,  
> I have a question regarding a global object.  
>  
>  
> I have an IDL process that will run independently on several  
> different computers. So, I need a method of deciding what  
> computer and operating system I'm on, and defining various  
> filepaths etc. So I have written an object with methods such  
> as "->getfilepath". This method determines the computer name, and  
> returns the appropriate info. There are several methods, for different  
> read and write paths, and different data sets. The actual information is  
> hardcoded into the object definition.  
>  
> So, I currently create the object and kill the object deep in each  
> basic read/write function, but I would much rather have one instance  
> of the object (i.e. the singleton object) that can be seen everywhere.  
> (I've used this approach many times in labview, and it is really handy).  
> I actually thought that objects were supposed to be of that scope, but  
> I guess I got confused with the Labview nonreentrant vis.  
>  
> So, is there a way to easily define the object, or do I simply create it  
> upon startup of my IDL and define it to a system variable?

Oh, oh. Pavel, where are you?

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438, E-mail: [david@dfanning.com](mailto:david@dfanning.com)

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

---

---

Subject: Re: global singleton object ?

Posted by [Mark Hadfield](#) on Tue, 02 Oct 2001 21:47:31 GMT

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

---

> So, is there a way to easily define the object, or do I simply create it  
> upon startup of my IDL and define it to a system variable?

Remember that functions are global in IDL.

Write a function that creates your settings object and assigns it to a system variable. On every entry to the function, check whether the system variable exists and points to a valid object. If it does, use that object.

I'm sure there are other ways to do this, and some of them may even be better than this one!

---

Mark Hadfield  
m.hadfield@niwa.cri.nz <http://katipo.niwa.cri.nz/~hadfield>  
National Institute for Water and Atmospheric Research

--

Posted from clam.niwa.cri.nz [202.36.29.1]  
via Mailgate.ORG Server - <http://www.Mailgate.ORG>

---

Subject: Re: global singleton object ?  
Posted by [R.G.S.](#) on Wed, 03 Oct 2001 13:11:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Mark Hadfield <m.hadfield@niwa.cri.nz> wrote in message  
news:004301c14b8b\$e33233e0\$d938a8c0@Hadfield...  
>> So, is there a way to easily define the object, or do I simply create it  
>> upon startup of my IDL and define it to a system variable?  
>  
> Remember that functions are global in IDL.  
>  
> Write a function that creates your settings object and assigns it to a  
> system variable. On every entry to the function, check whether the system  
> variable exists and points to a valid object. If it does, use that object.  
>  
> I'm sure there are other ways to do this, and some of them may even be  
> better than this one!  
>  
> ---  
> Mark Hadfield  
> m.hadfield@niwa.cri.nz <http://katipo.niwa.cri.nz/~hadfield>  
> National Institute for Water and Atmospheric Research

Yes, that is what I have done (a procedure actually).  
In every routine that I use the object, I first call a "object\_initialize"

procedure that starts the object if it does not already exist.  
That seems to be the way to go with using a "fake" singleton object.  
(of course, the main code destroys the object when it cleans up)

Cheers,  
bob

---

---

Subject: Re: global singleton object ?  
Posted by [Pavel A. Romashkin](#) on Wed, 03 Oct 2001 15:22:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Bob,

The following URL points to a tiny file that does what you want.  
[http://spot.colorado.edu/~romashki/idl/sobj\\_new.sav](http://spot.colorado.edu/~romashki/idl/sobj_new.sav)  
The SOBJ\_NEW does not call OBJ\_NEW is a singleton object already exists,  
thus avoiding OBJ\_NEW overhead.  
I also recommend checking out  
[http://spot.colorado.edu/~romashki/idl/single\\_set.sav](http://spot.colorado.edu/~romashki/idl/single_set.sav)  
which provides essentially the same services except the contents can be  
of any type, not just object. SINGLE\_SET does not loop through heap  
variables, fetching the only one (the named one) of interest.  
The actual code is too embarrassing to share, JD would not let me on the  
newsgroup again :-(

```
s = sobj_new(/help) ; How to use it
s = sobj_new() ; Make new object
d = sobj_new() ; Get a reference to it again
help, s, d ; Make sure they are the same
; Set NO_KILL to see what it does
s -> store, obj_new('IDLgrModel'), /no_kill
s -> store ; Take a look at contents...
d -> store ; They are the same
obj_destroy, s ; And kill it (by accident).
d = sobj_new() ; NO_KILL was set, so it did not die.
; But NO_KILL is NOT set in D now, so that heap-gc, if
; ever used, does not go into an infinite loop.
d -> store ; Make sure contents are the same
; except for NO_KILL setting. Set it again if you want.
```

Cheers,  
Pavel

P.S. The only use for a Singleton (or SINGLE\_SET) that I can see is  
avoiding Common blocks while

still allowing sharing some information between widget programs that may or may not be launched independently, without using linking arguments. Or, as people asked before, keeping a Preferences structure for the duration of a session. When I wrote Display routines, I had to use Common block to keep track of linked displays and the topmost one because they were intended to be used interactively, and there is no way to define how many of them are present and in which order. A singleton would allow this without a common block.

---

---

Subject: Re: global singleton object ?

Posted by [R.G.S.](#) on Wed, 03 Oct 2001 16:59:47 GMT

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

---

Pavel A. Romashkin <pavel.romashkin@noaa.gov> wrote in message  
news:3BBB2D52.346B882B@noaa.gov...

```
> Bob,
>
> The following URL points to a tiny file that does what you want.
> http://spot.colorado.edu/~romashki/idl/sobj_new.sav
> The SOBJ_NEW does not call OBJ_NEW is a singleton object already exists,
> thus avoiding OBJ_NEW overhead.
> I also recommend checking out
> http://spot.colorado.edu/~romashki/idl/single_set.sav
> which provides essentially the same services except the contents can be
> of any type, not just object. SINGLE_SET does not loop through heap
> variables, fetching the only one (the named one) of interest.
> The actual code is too embarrassing to share, JD would not let me on the
> newsgroup again :-(
>
> s = sobj_new(/help) ; How to use it
> s = sobj_new() ; Make new object
> d = sobj_new() ; Get a reference to it again
> help, s, d ; Make sure they are the same
> ; Set NO_KILL to see what it does
> s -> store, obj_new('IDLgrModel'), /no_kill
> s -> store ; Take a look at contents...
> d -> store ; They are the same
> obj_destroy, s ; And kill it (by accident).
> d = sobj_new() ; NO_KILL was set, so it did not die.
> ; But NO_KILL is NOT set in D now, so that heap-gc, if
> ; ever used, does not go into an infinite loop.
> d -> store ; Make sure contents are the same
> ; except for NO_KILL setting. Set it again if you want.
>
> Cheers,
> Pavel
>
```

- > P.S. The only use for a Singleton (or SINGLE\_SET) that I can see is
- > avoiding Common blocks while
- > still allowing sharing some information between widget programs that may
- > or
- > may not be launched independently, without using linking arguments. Or, as
- > people asked before, keeping a Preferences structure for the duration of a
- > session. When I wrote Display routines, I had to use Common block to keep
- > track of linked displays and the topmost one because they were intended to
- > be used interactively, and there is no way to define how many of them are
- > present and in which order. A singleton would allow this without a common
- > block.

Thanks for the reply!

As for why a singleton is useful, just wait til IDL is multithreaded. When you have multiple processes that need acces to information (info that can be changed by the user) then a singleton object is very nice! Very Very Nice.

Cheers,  
bob stockwell

PS BUFFY LIVES!

ack, geek cover blown, get out of that newsgroup now!

---

Subject: Re: global singleton object ?

Posted by [Pavel A. Romashkin](#) on Wed, 03 Oct 2001 17:36:12 GMT

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

---

- > As for why a singleton is useful, just wait til IDL is multithreaded. When
- > you
- > have multiple processes that need acces to information (info that can be
- > changed by the user) then a singleton object is very nice! Very Very Nice.

That probably will depend on what is allowed to be global between the threads. My singleton relies on a function being global, and heap memory addresses being global, too. Will each thread use separate heap addressing? Now, that and the synchronization of the threads is a separate can of worms that I have no knowledge about. It is almost the same as distributed computing, isn't it?

Pavel

---

---

Subject: Re: global singleton object ?

Posted by [Craig Markwardt](#) on Wed, 03 Oct 2001 19:27:02 GMT

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

---

"Pavel A. Romashkin" <pavel.romashkin@noaa.gov> writes:

> Bob,

>

> The following URL points to a tiny file that does what you want.

> [http://spot.colorado.edu/~romashki/idl/sobj\\_new.sav](http://spot.colorado.edu/~romashki/idl/sobj_new.sav)

> The SOBJ\_NEW does not call OBJ\_NEW is a singleton object already exists,  
> thus avoiding OBJ\_NEW overhead.

> I also recommend checking out

> [http://spot.colorado.edu/~romashki/idl/single\\_set.sav](http://spot.colorado.edu/~romashki/idl/single_set.sav)

Hey Pavel--

Is there any reason these have to be posted as SAV files? You saved  
it with IDL 5.5 which pretty much means you are catering to a \*very\*  
exclusive crowd right now ... :-)

Craig

--

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

---