

---

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.

---