
Subject: Re: Structures and COMMON blocks

Posted by [C.V.Nieuwenhuize](#) on Wed, 10 Apr 1996 07:00:00 GMT

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

Phil Williams <williams@irc.chmcc.org> wrote:

> I'm developing several widget based apps and have a question concerning
> the use of structures and common blocks.

> I have found both of these invaluable, but the problem comes when I want
> to add a variable to a common block or structure. When I try to
> recompile the .pro file idl won't let me. Is there another way to do
> this other than quitting IDL and starting it over?

> Along this theme, is there a way to "uncompile" a procedure or function
> so that IDL forgets about it?

> Thanks in advance,

> --

> /*****
> Phil Williams, Ph.D.
> Research Instructor
> Children's Hospital Medical Center "One man gathers what
> Imaging Research Center another man spills..."
> 3333 Burnet Ave. -The Grateful Dead
> Cincinnati, OH 45229
> email: williams@irc.chmcc.org
> URL: http://scuttle.chmcc.org/~williams/
> *****/

Well,

I've encountered the same problem. My solution:

Use a (fixed number of) anonymous structures in the common blocks, e.g.
for keeping the widget ID's. So when you decide to add a widget, you
can add a field in the (anonymous) structure without restarting IDL.

As far as I know it isn't possible to change the size of a common
block, so you have to plan the number of variables in advance (e.g.
one (anonymous structure) for Widget ID's, one for return values, one
for status values etc.

some excerpts from one of my widget-app's:

```
; the first (sub) routine in the file, defining the common block
pro sXRDSQLPR_INIT_READ, nrlines
COMMON XRDSQLPR_BLOCK, state, res, exitcode, linesread, LinesToRead, $
```

```

LineBuffer, maand_code, $
  last_baannum, baancount, jaar, beschr
.
.
.
; the main function
function XRDSQLPR, GROUP=Group, RESULT=result,
BESCHRIJVING=beschrijving, STARTPATH=startpath
COMMON XRDSQLPR_BLOCK
  wid = $
{main:-1L,Inlezen:-1L,afsluiten:-1L,bestand:-1L,omvang:-1L,jaar:-1L,bestandjaar:-1L,$
  baanvakcount:-1L,actie:-1L,regelselectie:-1L,regelcount:-1L, $
  knoppen:-1L}

IF N_ELEMENTS(Group) EQ 0 THEN GROUP=0

junk  = { CW_PDMENU_S, flags:0, name:" }

MAIN = WIDGET_BASE(GROUP_LEADER=Group, $
  COLUMN=1, $
  SPACE=5, $
  XPAD=5, $
  YPAD=5, $
  MAP=1, $
  TITLE='Lees SQL printbestand', $
  UVALUE='MAIN')
wid.main = MAIN

wid.inlezen = WIDGET_BASE(MAIN, $
  COLUMN=1, $
  FRAME=1, $
  MAP=1, $
  TITLE='displaybase', $
  UVALUE='Inlezen')
.
.

WIDGET_CONTROL, MAIN, /REALIZE
state= {wid:wid, path:startpath,filename:"",filestat:FS, $
filemask:'*.*'}
widget_control, state.wid.regelcount, sensitive =0

XMANAGER, 'XRDSQLPR', MAIN

result = res
return , exitcode
end

```

I hope these suggestions will help you in some way,

Cees van Nieuwenhuize
e-mail: C.v.Nieuwenhuize@net.HCC.nl
Gouda, Netherlands

Subject: Re: Structures and COMMON blocks
Posted by [Ken Knighton](#) on Wed, 10 Apr 1996 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Phil Williams <williams@irc.chmcc.org> wrote:

> I'm developing several widget based apps and have a question concerning
> the use of structures and common blocks.
>

> I have found both of these invaluable, but the problem comes when I want
> to add a variable to a common block or structure. When I try to
> recompile the .pro file idl won't let me. Is there another way to do
> this other than quitting IDL and starting it over?

You can add fields to anonymous structures using CREATE_STRUCT. You can not change the definition of a named structure.

As for common blocks, I don't know of any way to change a common block definition in the current IDL session. That means you have to exit IDL and get back in.

This might sound like a burden, but, except in special cases where you want to have a "static" value in a routine, you don't need to use common block variables. Instead, you can save information into an anonymous structure and then store the structure in a widget's uvalue and retrieve it at will. For examples of this, see the compound widgets supplied with IDL such as CW_FIELD.

If you really need to have a variable common block, then you can define a common block variable as a structure and simply add fields to the structure using CREATE_STRUCT.

```
IDL> common xyz_com, sGlobal
IDL> sGlobal = {a:5, b:'xyz'}
IDL> print, sGlobal
{    5 xyz}
IDL> sGlobal = CREATE_STRUCT(sGlobal, 'c', 7.5)
IDL> print, sGlobal
{    5 xyz    7.50000}
```

>

> Along this theme, is there a way to "uncompile" a procedure or function
> so that IDL forgets about it?

I am not aware of any.

I hope this helps.

Ken Knighton knighton@cts.com
San Diego, CA

Subject: Re: Structures and COMMON blocks
Posted by [plugge](#) on Tue, 16 Apr 1996 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <316ADC7A.41C6@irc.chmcc.org>, Phil Williams <williams@irc.chmcc.org> writes:
|>I'm developing several widget based apps and have a question concerning
|>the use of structures and common blocks.

|>
|>I have found both of these invaluable, but the problem comes when I want
|>to add a variable to a common block or structure. When I try to
|>recompile the .pro file idl won't let me. Is there another way to do
|>this other than quitting IDL and starting it over?

|>
Phil,
there is no way to enlarge a named common block; but there is a trick to do
it. I use often dummy variables at the end of a block (for example:

COMMON block, var1,var2,var3,v1,v2,v3,v4

where v1,v2,v3,v4 are (unused) dummy variables; if you need an additional
variable in the common block, you can just rename one of these dummies, without
need to exit the current IDL session.

Michel

Michael Plugge Fachhochschule Mannheim
Institute for Statistics / \ Hochschule fuer Technik und Gestaltung
and Image Processing / \ Email: plugge@biv7.sr.fh-mannheim.de
Speyerer Str. 4 / \ Tel: 0621 2926208
68163 Mannheim / \ --o ^
Germany / - \<,- / \ ^
 / () / () / \ ^
