Subject: Re: problem with program code area full in VMS IDL 4.0.1
Posted by davidf on Tue, 22 Jul 1997 07:00:00 GMT
View Forum Message <> Reply to Message

Martha Kusterer writes:

> Can anyone help me?
>
> I am running in IDL 4.0.1 on VMS and I am now getting program code area
> full error. I split the subroutine up into little moduals but it still
> gets the error. The whole program is a GUI to display satellite data and
> is quite complicated. From the IDL user documentation I don't get enough
> information to allow me to figure out what else to do. Does anyone have
> any ideas on this topic.
>
> I also tried the .SIZE command to enlarge the program data area and it
> doesn't seem to work. I realize that in IDL 5.0 this will no longer be a
> problem but we are limlited by our operating systme and I don't know
> when we will be able to update to it.

If you split the subroutine up into little moduals and it still
doesn't compile, then my guess is that something else is wrong.
Here is what I have seen happen from time to time, especially with
widget programs. And particularly in IDL 5, where it is difficult
to tell if there is a problem in a widget module.

There is a coding error in one of the modules. When you compile the
code, most of the modules compile, but this one particular one does
not. Yet when you run the code, the program appears to work. You can't
figure out what is wrong and you start to get nonsensical errors
("program data area full"!?).

Try this. Exit IDL. Come back in. Compile all of the modules
by using the .Compile command. Does every module compile?
Are you sure? Run the program. Does it work now?

I think that one of the (now!) many modules is not compiling
properly. Yet you have a (somewhat) working version of that
module in IDL's compiled memory space, so the program appears
to work. Each time you run the program, you are trying to create
variables in the context of the broken module instead of in the
context of the main IDL level.

Anyway, I think I can guarantee that if this isn't the problem,
it's probably something else. :-)

Cheers,

David

--- -
David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com
IDL 5 Reports: http://www.dfanning.com/documents/anomaly5.html

---

Subject: Re: problem with program code area full in VMS IDL 4.0.1
Posted by Stein Vidar Hagfors H on Fri, 25 Jul 1997 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Martha Kusterer writes:
>
>> Can anyone help me?
>>
>> I am running in IDL 4.0.1 on VMS and I am now getting program code area
>> full error. I split the subroutine up into little moduals but it still
>> gets the error. The whole program is a GUI to display satellite data and
>> is quite complicated. From the IDL user documentation I don't get enough
>> information to allow me to figure out what else to do. Does anyone have
>> any ideas on this topic.
>>
>> I also tried the .SIZE command to enlarge the program data area and it
>> doesn't seem to work. I realize that in IDL 5.0 this will no longer be a
>> problem but we are limlited by our operating systme and I don't know
>> when we will be able to update to it.
>
> If you split the subroutine up into little moduals and it still
> doesn't compile, then my guess is that something else is wrong.
> Here is what I have seen happen from time to time, especially with
> widget programs. And particularly in IDL 5, where it is difficult
> to tell if there is a problem in a widget module.

The following command also generates "% Program code area full" errors:

IDL> dummy = execute(too_long_string)

(where the "too_long_string" is a syntactically correct, but simply too
long
command string)

Some times programs are written to construct complicated statements to

be
executed by the execute() function, and when users come along and use
the
programs for more complicated scenarios, it breaks down like this.

So, look for "execute(..)"-statements in the code, and try inserting
e.g.,

```
print,"Before exec"
.... = execute(....)
print,"After exec"
```

at all those places, and then recompile. If the output is

```
Before exec
% Program code area full
After exec
```

then you've found the cause. Now, simply rewrite the offending code :-)

(A hint: Often, such execute statements are used to make anonymous
structures with variable number/names of tags - nowadays the routine
create_struct may be used to do this. I think the create_struct
was introduced in IDL after e.g., version 3.0, and prior to that one
had to use execute-statements...)

Stein Vidar
