Subject: Re: GUI interface update issues
Posted by Haje Korth on Tue, 21 Jun 2005 18:23:20 GMT
View Forum Message <> Reply to Message

Rick,
Yes trying everything on unix takes time and all you will get out of it is a little more understanding on how IDL is coded internally. I thought the same spook that causes the IDLDE to be unresponsive is the origin of your update problem. (I still want the unix-like command line version in Windows, and damn it, I forgot to put that on the final beta questionaire.)

Haje

"Rick Towler" <rick.towler@nomail.noaa.gov> wrote in message news:d99l4c$ua5$1@news.nems.noaa.gov...
>
>
> Haje Korth wrote:
>> Rick,
>> my question would be whether the same things happens on UNIX OS. On
>> Windows I have found IDL has the terrible habbit of taking over with
>> unpredictable results such as GUIs not reacting at all. If the UNIX
>> version works fine, which I suspect it will, your only hope is some input
>> from the developers.
>
> Hey Haje,
>
> Well, I suppose I could install the IDL VM on a linux box and compile
> everything... Doesn't help me too much though if I find it's o.k. on
> linux. I too have had issues with the windows GUI but usually it is that
> IDLDE becomes unresponsive but my applications are o.k.
>
> Upon further investigation I have determined that the dlm angle is wrong.
> The issue occurs with or without calling the dlm but it only occurs with
> one of the many models in the application. While the calculations are
> basically the same, that model has an additional nested for loop in the
> IDL code.
>
> <sigh>
>
> -Rick
>
>
>
>>
>> "Rick Towler" wrote in message news:d99f2fohk1@news.nems.noaa.gov...
>>

>>> Hello group,
>>>
>>> I have an interesting problem with gui redraw on WinXP which maybe
>>> someone can shed some light on.
>>>
>>> I have an application that uses David's progress bar object. We have
>>> some potentially long calculation times and it is important to give a
>>> little feedback. It worked great until I added the ability to call a new
>>> calculation routine written in C as a dlm. If I run the application the
>>> old way (all functions written in IDL) the progress bar and GUI interface
>>> function normally. But if I run the application using the external
>>> routine eventually the GUI interface stops redrawing, the drop down menu
>>> text disappears, and the progress bar fails to update. The application
>>> runs and when calculations are finished it returns to normal (progress
>>> bar is destroyed and interface works as expected) but there is no
>>> feedback while running. A problem since the new calculations can take
>>> hours and hours and it is nice to see where it is in the process.
>>>
>>> The program is structured like so:
>>>
>>> ;-------------
>>> setup stuff
>>>
>>> for loop begin
>>>
>>>   progressBar->update
>>>
>>>   if (use_dlm) then begin
>>>   myDLM_proc, param1, param2, OUT1=out1, OUT2=out2...
>>>   endif else begin
>>>   IDLbased_proc, param1, param2, OUT1=out1, OUT2=out2...
>>>   endelse
>>>
>>>   for loop begin
>>>
>>>   if (use_dlm) then begin
>>>   anotherDLM_proc, param1, param2, OUT1=out1,OUT2=out2...
>>>   endif else begin
>>>   anotherIDLbased_proc, param1, param2, OUT1=out1,OUT2=out2...
>>>   endelse
>>>
>>>   endfor
>>>
>>> endfor
>>> ;------------
>>>
>>> Actually a lot more is going on but you get the idea.
>>>

>>> And to elaborate on what I mean by "eventually the gui stops responding".
>>> If I start the application and run a short calculation (~2 minutes) the
>>> first time the gui functions normally. But with every subsequent run the
>>> progress bar moves maybe 20% of the way then I lose the gui. If I close
>>> and restart the application the same thing happens, first one works, then
>>> problems). If I run a long calculation (hours) the progress bar never
>>> moves past the first tick (maybe 1%).
>>>
>>> Why does IDL stop updating the gui? Any ideas? While the functions in
>>> the dlm are compute intensive, they aren't particularly complicated and
>>> only rely on some simple macros in an .h file. Just a *lot* of looping
>>> over a moderate amount of data.
>>>
>>> -Rick
>>
>>