Subject: Re: Object Graphics when using 'stop'
Posted by David Fanning on Sat, 11 May 2002 19:39:49 GMT
View Forum Message <> Reply to Message

Randall Skelton (rhskelto@atm.ox.ac.uk) writes:

> Well this is certainly not good news... The whole point of me using OG was
> so that I could actively pan-and-scan through a very complex data array.
> Personally, I need to be able to do this when I use STOP or set
> breakpoints.
>
> Surely all those die-hard proponents of IDL OG will be hard at work this
> weekend trying to come up with a fix for this ;)

I have long thought of myself as Cassandra, crying in the wilderness, desperate to get across my message of the power of prayer. But my words continue to fall on deaf ears. (We even have a new call for a "wish list" this week.) People! Wake up! The gods listen to our prayers and answer them. The result is always chaos and confusion.

This whole business, I'm afraid, is because some people (I won't name them, but you would recognize the names, I'm sure) would not be content running widget programs as the gods intended them to be run: one at a time. No, they wanted to run multiple widget programs AND type at the IDL command line, all at the same time. So, we were given this, and it has all been downhill since.

So, whatever it is that STOPS working with a STOP command, it is the thing that makes widgets work while the interpreter is working, too. (I intend to wave my hands here a bit, because I don't understand the details, or even the fundamentals. I'm just going to tell you how to work around whatever it is.) Let's just say there are loops within loops, and leave it at that. When you STOP whatever it is, call it the "initial interpreter", you stop the ability to send widget events to widget programs from that same "event loop". (This was all discussed the other day.)

What we would like to do at the moment we are stopped, is start up *another* event loop that could work on our widget program, while the initial interpreter is momentarily stopped. We can do this, if we make the new widget program a BLOCKING widget program.

Now, I don't want to go far into the details of this, but recall that only ONE program can BLOCK. If you try to run two blocking programs, only the first one blocks the command line.

So, if the program you are running is a non-blocking program.
(It sounds like it to me.) Then you can run an object graphics
widget program, IF you can get this widget program to run as
a blocking widget program.

I've added a BLOCK keyword to FSC_SURFACE so you can run
it as a blocking program if this keyword is set.

   http://www.dfanning.com/programs/fsc_surface.pro

In your little example test program, I can stop the program,
then call FSC_SURFACE like this:

```
   IDL> FSC_SURFACE, data
```

Not only does the surface show up, but I can rotate the surface,
print, make JPEG files, etc. I have to close the program before
I can type .CONTINUE in the stopped program.

> PS: David, on what page of your book is this bizarre IDL widget behavior
> described?

Oh, I'm sure it's described somewhere. I just can't find it. :-(

> Surely you have an insightful comment to add?

Of course.

The secret to a happy life is not getting what you
want. It is wanting what you get.

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155