
Subject: Arg_Present, XSurface, and Other Assorted Blunders

Posted by [davidf](#) on Thu, 07 Aug 1997 07:00:00 GMT

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

Hi Folks,

Just a couple of days ago I was winching and moaning (as my friends in England say) about Arg_Present and how its name was just a bit misleading, and how it would lead to programming errors by people who don't read the documentation, etc., etc.

Well, don't you know, that dog don't hunt far from home!

I have been getting infrequent, but persistent, reports from some of the people who downloaded my new graphics object program XSURFACE from my web page that it seemed to hang their IDL session. I could never reproduce the problem, so I just chalked it up to the usual whimsical programming gods having their little joke.

But today I did a clean install of IDL 5.0.2 on my Mac (I heard it was piling up outside of RSI waiting for the UPS folks to settle their differences, and sure enough it was there waiting for me when I went up to check out the rumor) and don't you know, the darn program hung up on me.

(Well, first it wouldn't run at all, because the default Mac IDL installation doesn't put **any** IDL directories on the path. But that is another story, and easily fixed if you have a firm conviction that every single one of your programs can't possibly have syntax errors on **every** single IDL library function call!)

Anyway, since it hung on the Mac I thought I would try it on my WindowsNT machine, so I hurried and installed IDL 5.0.2 there too. When I ran XSURFACE there, it hung too! But, wait a minute, I hadn't added my program libraries to the path (the IDL libraries **were** there on my WindowsNT install, thank goodness). Add my libraries, run it again, it works! Hummm...

The only library call I make is to something called LoadData. I looked at the code near the LoadData call and here is what I found:

Catch, error

```
IF error NE 0 THEN BEGIN ; Can't find LoadData.  
  data = DIST(41)  
  x = Findgen(41)  
  y = Findgen(41)  
ENDIF
```

```
IF Arg_Present(data) EQ 0 THEN BEGIN  
  data = LoadData(2)  
ENDIF
```

Here is what this piece of code is **meant** to do:
I want to supply some default data if the user doesn't pass data into the program in the argument "data".
If I need to create some data I want to use my LoadData program to get it. But I also know that some people won't have LoadData, so I want to "Catch" the error that happens when you try to call a program that is not on your path.
If I catch the error, I create the "data" variable and continue program execution.

Because this is a program using IDL 5 specific functionality, I also wanted to use the new Arg_Present routine to prove that I am an up-to-date and with-it IDL programmer.
Apparently, I forgot to read the documentation. In any case, here is what happens.

When the call is first made, "data" is not present and Arg_Present reports this correctly. The LoadData error occurs and I bounce up to my error handler. I define "data" and continue. But Arg_Present **STILL** reports data as missing in action. This is so even when it is defined AND a variable that is passed by reference. As a result, my code goes into an infinite loop.

So now I am confused about exactly what Arg_Present is suppose to do, but I do know this: it is a grievous mistake to treat Arg_Present as an function that tells you if an argument is present! To fix the problem I swallowed my pride and went back to the terribly misnamed N_Elements to solve my problem.

If that was **all** I had done, perhaps I could be forgiven. But in addition I broke one of the cardinal rules of programming error handlers. I forgot to **always, always** include some kind of error message in the handler.

If you don't, you will end up blaming RSI for something that is entirely (well, **almost** entirely) your own fault!

Anyway, some of you want to know how you get to be a renowned teacher of IDL programming. This is how you do it. Make lots and lots of foolish programming mistakes. And then when you get up in front of a class, try hard to remember what it feels like to stand there with egg all over your face. :-)

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: Arg_Present, XSurface, and Other Assorted B

Posted by [wmc](#) on Tue, 12 Aug 1997 07:00:00 GMT

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

In article e54555e6d6bd26f9896d3@news.frii.com, davidf@dfanning.com (David Fanning) writes:

```
> Catch, error
> IF error NE 0 THEN BEGIN ; Can't find LoadData.
>   data = DIST(41)
>   x = Findgen(41)
>   y = Findgen(41)
> ENDIF
>
> IF Arg_Present(data) EQ 0 THEN BEGIN
>   data = LoadData(2)
> ENDIF
```

> here is what happens.

```
> When the call is first made, "data" is not present and
> Arg_Present reports this correctly. The LoadData error occurs
> and I bounce up to my error handler. I define "data" and
> continue. But Arg_Present *STILL* reports data as missing
> in action. This is so even when it is defined AND a variable
> that is passed by reference. As a result, my code goes into
> an infinite loop.
```

Um, forgive me here, since I'm still an IDL 4 person... but it looks like arg_present is behaving quite correctly in this case. arg_present is supposed

to tell you if "data" is passed as an argument, no? And it isn't, even when you define it within the error routine. Or have I missed something here?

- William, "he's still on 4.0" (but our system people are working on it)

William M Connolley | wmc@bas.ac.uk | <http://www.nbs.ac.uk/public/icd/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself
