
Subject: DLL, IEEE question

Posted by [karl](#) on Tue, 01 Apr 1997 08:00:00 GMT

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

I'm having a very strange problem with linkimage and DLL's. My DLL contains a call to a function from the IMSL library that is linked when the DLL is built. What seems to be happening is that somehow the internal IDL function that checks for IEEE infinities and NaN's gets mangled when I use linkimage to link my DLL.

Here's a simple example of what happens:

```
IDL> print,1.0/0.0
% Program caused arithmetic error: Floating divide by 0
% Detected at $MAIN$
      INF
IDL>linkimage,'my_func','my_func.dll'
IDL> print,1.0/0.0
```

IDL dies here

This is all happening under NT 4.0 and I'm using IMSL library functions that come with microsoft powerstation. Another interesting clue is that I can apparently "fix" the problem by generating an underflow after executing the linkimage and before generating any infinities, i.e. somehow the underflow turns the internal IDL IEEE checking back on (or something). This is a kludge that is obviously not a long term solution, so my question is - has anybody had this problem and solved it and/or does anybody know how I can learn more about the IDL internal IEEE functions (at least enough to try and figure out what's going on here) ? Thanks in advance for any tips,

- KY

PS - I appreciate any responses but please don't reply with "Make sure your code doesn't generate infinities" , i.e. the whole point of the IEEE standard is to be able to trap and respond to such things; it is in fact in an IDL function that my program calls that the divide by 0 occurs...

--

Karl Young Phone: (415) 750-2158 lab
UCSF (415) 750-9463 home
VA Medical Center, MRS Unit (114M) FAX: (415) 668-2864
4150 Clement Street Email:karl@gojira.berkeley.edu
San Francisco, CA 94121
