
Subject: function in function

Posted by nobody@nowhere.com (S on Thu, 11 Oct 2001 20:31:53 GMT

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

I wrote a function in my IDL program which opens a proprietary file format and dumps the data into an array, as I had to do it many times, I wrote this as a function. In this function, I find I had to convert fixed-point binary numbers to floats, so I wrote a second function which does this. It is called within the first function. It all works fine, but when I first start IDL, and compile, the program fails with an undefined reference to the second function. After the second compile, it works fine. I inserted the FORWARD_FUNCTION statement in the 1st function, and then the problem went away. I know that if I compile a procedure, with references to other procedures, all will work well. So I'm wondering why the function in a function gives this problem or if others have encountered this? The IDL documentation says that the FORWARD_FUNCTION statement merely helps IDL to decide if the call is a function or an array, but it is only needed in the second function, so I don't think it explains the observed behavior. At first I thought this was having something to do with the order in which the procedures are compiled, e.g. you shouldn't compile a procedure that makes reference to another procedure/function that has yet to be defined. But I think I would be hard pressed to guarantee that all the miles of code I've written would actually observe this rule, yet they all work just fine. So is there something different about functions vs. procedures that gives this effect?

--

Steve S.

steve@NOSPAMmailaps.org
remove NOSPAM before replying
