
Subject: Re: Compile Problem with IDL5.2
Posted by [davidf](#) on Thu, 22 Jul 1999 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Joerg Mosthaf (joerg.mosthaf@urz.uni-heidelberg.de) writes:

> Well, thanks for answering so fast, but this is probably not the cause of the
> problem, as I can see in the status-windows that all routines, including the
> offending one, are compiled. At least I presume that when IDL says
> % Compiled module: PatData
> that it really did that and compiled the function. But still it sometimes
> cannot find said routine when executing.
> I will try the suggestion in the second posting and report back.

Here is an illustration of what is going on for you.

Here is my program file, called "main.pro"

```
*****
```

```
PRO utility  
value = myfunction(5)  
END
```

```
FUNCTION myfunction, number  
RETURN, Randomu(seed, number)  
END
```

```
PRO main  
utility  
print, myfunction(3)  
END
```

```
*****
```

When I run this program, I get this:

```
IDL> main  
% Compiled module: MAIN.  
% Variable is undefined: MYFUNCTION.  
% Execution halted at:  UTILITY          2 C:\RS\IDL52\DAVID\main.pro  
%                   MAIN          10 C:\RS\IDL52\DAVID\main.pro  
%                   $MAIN$
```

This happens even if I compile the program first:

```
IDL> .compile main  
% Compiled module: UTILITY.  
% Compiled module: MYFUNCTION.  
% Compiled module: MAIN.
```

```
IDL> main
% Variable is undefined: MYFUNCTION.
% Execution halted at:  UTILITY      2 C:\RSI\IDL52\DAVID\main.pro
%                MAIN      10 C:\RSI\IDL52\DAVID\main.pro
%                $MAIN$
```

The problem here is that the first module, UTILITY, is calling the function, MYFUNCTION, that isn't compiled until *after* UTILITY is compiled. As a result, "myfunction" has been registered with IDL as a variable.

The problem is easily solved (without resorting to FORWARD_FUNCTION) by moving the function ahead of the UTILITY procedure in the file:

```
*****
FUNCTION myfunction, number
RETURN, Randomu(seed, number)
END

PRO utility
value = myfunction(5)
END

PRO main
utility
print, myfunction(3)
END
*****
```

Now, when I call it:

```
IDL> main
% Compiled module: MAIN.
    0.383502  0.653919  0.0668422
```

Cheers,

David

--

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