Subject: IDL 6.1 create_struct() peculiarity
Posted by Justin Bronn on Tue, 31 May 2005 20:00:50 GMT
View Forum Message <> Reply to Message

So I'm working with an existing piece of software that uses the execute() function to resolve structures:

```
IDL> sname = 'xosimple'
IDL> stmt = 'struct = {' + sname + '}'
IDL> struct = execute(stmt)
```

I want to get rid of the execute() call because it is not allowed when running the VM (a nasty little error message will greet the VM user). I thought I was smart, and I found a way around this, by using the create_struct() function instead:

```
IDL> struct = create_struct(NAME=sname)
```

Now while this seems to work great at the IDL command line, when I try to embed this method into a callable procedure, the create_struct() throws a strange warning message (but doesn't stop the execution of the code):

```
% CREATE_STRUCT: Unexpected keyword cleanup stack found on return.
```

When trying this method out on IDL 5.6, I get the following error message:

```
% CREATE_STRUCT: Incorrect number of arguments.
```

So here's my stab at what I think is going on: RSI decided to allow the create_struct() method to be called with only the NAME keyword set in IDL 6.0, however they forgot to do all the testing so it doesn't clean up the C keyword arguments properly when called in a non-interactive mode.

Below is the code for a procedure that embeds this create_struct() method, and the sample structure definition file that should re-create this issue (FYI - I'm using IDL 6.1.1 on Linux - Fedora Core1).

Should I worry about this warning message? If anyone has any insight, please enlighten me & the newsgroup :)

Thanks,
-Justin

;; init_structs.pro

```
;;
;; Routine that will initialize structures that have
;; their definition in a certain filename.  Needs the
;; definition files to be in the same directory.

pro init_structs

  files = file_search('./xo*__define.pro', COUNT=count)
  if count eq 0 then $
    message, 'Could not find any structure definition files!'

  for i = 0L, count-1 do begin
     strings = stregex(files[i], '(xo.*)__define.pro' $
                       , /EXTRACT, /SUBEXPR)
     struct_name = strings[1]
     struct = create_struct(NAME=struct_name)
  endfor

end

;; xosimple__define.pro
;;
;; Simple structure definition file

pro xosimple__define

  temp = {xosimple, $
          title:'', $
          int_value:0, $
          float_value:0.0 $
          }

end
```