Subject: Re: Is System Dependent Compilation Possible?
Posted by Thomas A. McGlynn on Thu, 25 Apr 1996 07:00:00 GMT
View Forum Message <> Reply to Message

```
plugge@biv6.sr.fht-mannheim.de wrote:
>
> In article <317EE71F.31D2@raptor.lpl.arizona.edu>, Tim Patterson
<tim@raptor.lpl.arizona.edu> writes:
> |>I have a tool written in IDL/Fortran and C that needs to
> |>run on both Unix and VMS systems. I have a few system
> |>dependent IDL procedures that I use as wrappers for the
> |>CALL_EXTERNALS.
> |>

.... ideas that didn't work snipped.

> |>Anybody got any ideas?
> |>
> |>     Thanks
> |>
> |>           Tim
> |>
>
> Tim,
> the problem is that the executive commands (for example, .run etc) cannot
> be called from normal program code. You should make the distinction in the
> procedure body, for example:
>
> pro wrapper
>   IF !version.os EQ 'vms' then begin
>     print,"wrapper: VMS version"
>     (VMS code goes here)
>   endif else begin
>     print,"wraper: UNIX version"
>     (UNIX code goes here)
>   endif
> end
>
> best regards
>
> Michel
>
>  ------------------------------------------------------------ -------------
> Michael Plugge              _      Fachhochschule Mannheim
> Institute for Statistics   / \     Hochschule fuer Technik und Gestaltung
> and Image Processing      /   \    Email:  plugge@biv7.sr.fh-mannheim.de
> Speyerer Str. 4          /     \   Tel:    0621 2926208
> 68163 Mannheim          /       \              --o       /\
>  Germany              /         --             - \<,-      /  \/\
>                    /            \           (_)/ (_)    /      \/\
>   ----------------------------------------------------------- -------------
```

If you really need separate files you could use the following technique.
It's a bit of a kludge but not too bad.

Put your machine dependent code in files which you suffix with the
machine name and create a dummy procedure at the end of the
file with the same name as the file. Then you can do:

```
   IF !version.os EQ 'vms' then begin
       procedure_vms
   ENDIF ELSE BEGIN
       procedure_unix
   endelse

   ; Now the correct version of procedure is compiled and I can use
   ; it as I wish.
```

where procedure_vms.pro would look like:

```
   pro procedure .....    ; The system dependent procedure or function to
                          ; be compiled.
   ....                   ; Body of procedure
   end
   ....                   ; Any associated system-dependent
     ; procedures you want in the same file.

   pro procedure_vms      ; A dummy procedure with no body or arguments.
   end
```

One benefit of this is that if the system dependent procedure calls
a lot of system dependent sub-procedures (or functions), you can put the
primary procedure first since the whole file will get processed as long as
the dummy procedure is last. I hate having to put all the helper functions
in my large procedures first. It seems counterintuitive and makes
it more difficult to follow the program flow.

I confess I've never actually used this technique in production code, but
it should work ok.

Good luck,
Tom McGlynn
tam@silk.gsfc.nasa.gov
HEASARC

Subject: Re: Is System Dependent Compilation Possible?
Posted by Tim Patterson on Thu, 25 Apr 1996 07:00:00 GMT

View Forum Message <> Reply to Message

Thanks for the ideas, guys

It seems that the RESOLVE_ROUTINE may be what I need.
I'll have to check that all the users are up-to-date with their
IDL versions.

Making the branch in the IDL code itself would work
for a small number of routines. Unfortunately I
have to have wrappers for around 50+ Fortran routines
and it becomes very tedious editing every routine
in that manner :)

Tim

Subject: Re: Is System Dependent Compilation Possible?
Posted by Robert Moss on Thu, 25 Apr 1996 07:00:00 GMT

View Forum Message <> Reply to Message

Tim Patterson wrote:
>
> I have a tool written in IDL/Fortran and C that needs to
> run on both Unix and VMS systems. I have a few system
> dependent IDL procedures that I use as wrappers for the
You may want to consider using RESOLVE_ROUTINE, which allows you to
compile a procedure from within a procedure or function.

>
>   IF !version.os EQ 'vms' then .run vms_version.pro
>
> produces an error.
>
> Anybody got any ideas?
>
>   Thanks
>
>   Tim

--
Robert M. Moss, Ph.D. - mossrm@texaco.com
 --- -
This does not necessarily reflect the opinions of Texaco Inc.

Subject: Re: Is System Dependent Compilation Possible?
Posted by Jackel on Thu, 25 Apr 1996 07:00:00 GMT

View Forum Message <> Reply to Message

In article <317EE71F.31D2@raptor.lpl.arizona.edu> Tim Patterson <tim@raptor.lpl.arizona.edu> writes:

```
> I have a tool written in IDL/Fortran and C that needs to
> run on both Unix and VMS systems. I have a few system
> dependent IDL procedures that I use as wrappers for the
> CALL_EXTERNALS.
> What I would like to do is have a batch file, say
> compile.pro, which can be invoked with @compile.pro
> and will compile the correct version of wrappers
> depending upon the OS, e.g.
> Anybody got any ideas?
>         Thanks
>                 Tim
```

If you've got a recent version of IDL, take a look at RESOLVE_ROUTINE. From the help file:

> The RESOLVE_ROUTINE procedure compiles user-written or library procedures or functions, given their names. Routines are compiled even if they are already defined. This procedure is similar to the .COMPILE executive command, but can be invoked within procedures and functions.
>
> Calling Sequence: RESOLVE_ROUTINE, Name

Hope this helps

Brian

Subject: Re: Is System Dependent Compilation Possible?
Posted by plugge on Thu, 25 Apr 1996 07:00:00 GMT

View Forum Message <> Reply to Message

In article <317EE71F.31D2@raptor.lpl.arizona.edu>, Tim Patterson <tim@raptor.lpl.arizona.edu> writes:

```
|>I have a tool written in IDL/Fortran and C that needs to
|>run on both Unix and VMS systems. I have a few system
|>dependent IDL procedures that I use as wrappers for the
|>CALL_EXTERNALS.
|>
|>What I would like to do is have a batch file, say
|>compile.pro, which can be invoked with @compile.pro
|>and will compile the correct version of wrappers
```

```
|>depending upon the OS, e.g.
|>
|>
|>IF !version.os EQ 'vms' then begin
|>
|> .run vms_version.pro
|>
|>ENDIF ELSE BEGIN
|>
|> .run unix_version.pro
|>
|>ENDELSE
|>
|>
|>Unfortunately, I can't seem to come up with anything
|>that even approaches this.
|>
|> IF !version.os EQ 'vms' then .run vms_version.pro
|>
|>produces an error.
|>
|>
|>Anybody got any ideas?
|>
|> Thanks
|>
|>  Tim
|>
```

Tim,
the problem is that the executive commands (for example, .run etc) cannot be called from normal program code. You should make the distinction in the procedure body, for example:

```
pro wrapper
  IF !version.os EQ 'vms' then begin
    print,"wrapper: VMS version"
    (VMS code goes here)
  endif else begin
    print,"wraper: UNIX version"
    (UNIX code goes here)
  endif
end
```

best regards

Michel

```
 ------------------------------------------------------------ -------------
Michael Plugge               _      Fachhochschule Mannheim
Institute for Statistics   / \     Hochschule fuer Technik und Gestaltung
and Image Processing      /   \    Email:  plugge@biv7.sr.fh-mannheim.de
Speyerer Str. 4          /     \   Tel:    0621 2926208
68163 Mannheim          /       \                 --o        /\
Germany                /         --             - \<,-      /  \/\
                      /            \          (_)/ (_)    /      \/\
 ------------------------------------------------------------ -------------
```