Subject: Re: check for duplicate routine names?
Posted by davidf on Fri, 19 Jan 2001 14:25:57 GMT
View Forum Message <> Reply to Message

Michael W Asten (masten@mail.earth.monash.edu.au) writes:

> It is a stupid error to have a two routines of the same name
> DoitNow.pro , in two different library files MyLib1.pro and MyLib2.pro
> .
> But your correspondent is sometimes stupid. The consequence of course
> is that the second compilation (of MyLib2.pro) over-rides the first (of
> MyLib1.pro) , so whatever was intended by maintenance of the code in
> MyLib1.pro does not execute. Or worse, when the two libs are compiled
> in different sequences, results become unpredictable.
>
> IDL does not appear to give any warning of the compilation of a routine
> of same name as one already compiled. Has anyone any ideas on how such
> errors can be trapped ?
>
> I suggest that at least in an IDL Project, the command "Compile all
> routines" should include such checking as a debugging tool. Or am I
> merely shifting blame for my stupidity onto the compiler? (you know how
> it is when you bang your head the second time, on the same kitchen
> cupboard door?).

What you need is a language with a little more
structure to it. Why not give up IDL and try
Pascal. :-)

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

Subject: Re: check for duplicate routine names?
Posted by davidf on Fri, 19 Jan 2001 14:47:17 GMT
View Forum Message <> Reply to Message

David Fanning (davidf@dfanning.com) writes:
>
> What you need is a language with a little more

> structure to it. Why not give up IDL and try
> Pascal. :-)

Whoops! Michael, that maybe came across harder than I
intended. And I'm already fearing another "rude" message
from Rochelle. :-(

What I was trying to say is that IDL is a "loose" sort
of language. That gives us fits sometimes, but it is
also what gives us so much power. Yes, IDL could be
doing all kinds of checking of all kinds of things.
It would be substantially slower, substantially less
powerful, and probably cost substantially more. And
it would be a hell of a lot less fun to work with.

You will smack your head a couple or three times.
But even the dullest of us (I speak here from
experience) eventually learn to duck. :-)

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

Subject: Re: check for duplicate routine names?
Posted by Jaco van Gorkom on Fri, 19 Jan 2001 14:54:13 GMT

View Forum Message <> Reply to Message

David Fanning wrote:

> Michael W Asten (masten@mail.earth.monash.edu.au) writes:
>
>> IDL does not appear to give any warning of the compilation of a routine
>> of same name as one already compiled. Has anyone any ideas on how such
>> errors can be trapped ?

> What you need is a language with a little more
> structure to it. Why not give up IDL and try
> Pascal. :-)
>
> Cheers,

>
> David

Now that reply came a bit too easy, I would say. Even IDL has quite some
structure to it, and this is definitely one of the things which *could*
be checked, if maybe only at runtime. ROUTINE_INFO,/SOURCE provides the
full path to the origin of any compiled routines, so a simple check
there could warn for "recompilation from a different origin".

Now to write this kind of check yourself you would need a hook onto when
IDL starts automatic compilation...

cheers,
Jaco

Subject: Re: check for duplicate routine names?
Posted by dominik on Fri, 19 Jan 2001 15:10:06 GMT
View Forum Message <> Reply to Message

>>>> > "MWA" == Michael W Asten <masten@mail.earth.monash.edu.au> writes:

MWA> It is a stupid error to have a two routines of the same name
MWA> DoitNow.pro , in two different library files MyLib1.pro and MyLib2.pro
MWA> .
MWA> But your correspondent is sometimes stupid. The consequence of course
MWA> is that the second compilation (of MyLib2.pro) over-rides the first (of
MWA> MyLib1.pro) , so whatever was intended by maintenance of the code in
MWA> MyLib1.pro does not execute. Or worse, when the two libs are compiled
MWA> in different sequences, results become unpredictable.

MWA> IDL does not appear to give any warning of the compilation of a routine
MWA> of same name as one already compiled. Has anyone any ideas on how such
MWA> errors can be trapped ?

If you are using Emacs with IDLWAVE, it has tools to catch such
cases. See the manual and a recent thread where JD explains how to
use it for detecting load-path shadows.

- Carsten

```
--
Carsten Dominik <dominik@astro.uva.nl>          \ _ /
Sterrenkundig Instituut "Anton Pannekoek"        |X|               _
Kruislaan 403; NL-1098 SJ Amsterdam             /| |\   _  _     _/ \
phone +31 (20) 525-7477; FAX +31 (20) 525-7484 __|o|___/ ~~ \___/    ~~~
```

Subject: Re: check for duplicate routine names?
Posted by Paul van Delst on Fri, 19 Jan 2001 16:23:30 GMT

View Forum Message <> Reply to Message

David Fanning wrote:

```
>
> David Fanning (davidf@dfanning.com) writes:
>>
>> What you need is a language with a little more
>> structure to it. Why not give up IDL and try
>> Pascal. :-)
>
> Whoops! Michael, that maybe came across harder than I
> intended. And I'm already fearing another "rude" message
> from Rochelle. :-(
```

It does seem of late that there are a number of new(er) posters who appear unfamiliar with the ways
of the Coyote. :o)

paulv

```
--
Paul van Delst                A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP               Drink deep, or taste not the Pierian spring;
Ph:   (301) 763-8000 x7274    There shallow draughts intoxicate the brain,
Fax:  (301) 763-8545          And drinking largely sobers us again.
Email: pvandelst@ncep.noaa.gov                 Alexander Pope.
```

Subject: Re: check for duplicate routine names?
Posted by Craig Markwardt on Fri, 19 Jan 2001 16:34:11 GMT

View Forum Message <> Reply to Message

Michael W Asten <masten@mail.earth.monash.edu.au> writes:

```
> It is a stupid error to have a two routines of the same name
> DoitNow.pro , in two different library files MyLib1.pro and MyLib2.pro
> .
> But your correspondent is sometimes stupid.  The consequence of course
> is that the second compilation (of MyLib2.pro ) over-rides the first (of
> MyLib1.pro) , so  whatever was intended by maintenance of the code in
> MyLib1.pro does not execute.  Or worse, when the two libs are compiled
> in different sequences, results become unpredictable.
```

Yes, I think IDL would be improved by more quality-of-code features
that could detect problems like this.

Another thing that would be nice is warnings about ambiguous uses of
variables that might shadow a function in the library. Note this
code:

```
max = max(x, min=min)
```

The next time you use min(x), will you get the function or the
variable named min?

You might say, "oh, I used square brackets, this is not a problem,"
but since round parenthesis are still allowed (unless you disable
them), I believe this can still be a problem. This has bitten me a
few times, so I'm not saying it's a huge deal. It is a compile time
issue since peoples' function libraries are different.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: check for duplicate routine names?
Posted by Pavel A. Romashkin on Fri, 19 Jan 2001 16:42:01 GMT

View Forum Message <> Reply to Message

If I had a vote, I'd be strongly against such checking.
This means that any time I try to compile a routine which is already
compiled, I'd be stopped by a warning message?! That would be a disaster
for developing the code.
My take on it, avoid routines with generic names. It is tempting to
write a program called "Just_Doit.pro", but it pays to add
"prj1_just_doit.pro", unless you are writing a *library* function that
will be used universally by your other projects. In that case, there
better not be more than one version. I use these prefixes all the time,
and it gives some side benefits, too (less so now with Projects
available) - sorting by name in file manager puts all related files
together, etc.

Cheers,
Pavel

Michael W Asten wrote:
>
> It is a stupid error to have a two routines of the same name

> DoitNow.pro , in two different library files MyLib1.pro and MyLib2.pro
> .
> But your correspondent is sometimes stupid. The consequence of course
> is that the second compilation (of MyLib2.pro) over-rides the first (of
> MyLib1.pro) , so whatever was intended by maintenance of the code in
> MyLib1.pro does not execute. Or worse, when the two libs are compiled
> in different sequences, results become unpredictable.
>
> IDL does not appear to give any warning of the compilation of a routine
> of same name as one already compiled. Has anyone any ideas on how such
> errors can be trapped ?
>
> I suggest that at least in an IDL Project, the command "Compile all
> routines" should include such checking as a debugging tool. Or am I
> merely shifting blame for my stupidity onto the compiler? (you know how
> it is when you bang your head the second time, on the same kitchen
> cupboard door?).
>
> Regards,
> Michael Asten

Subject: Re: check for duplicate routine names?
Posted by Liam E. Gumley on Fri, 19 Jan 2001 16:52:51 GMT

View Forum Message <> Reply to Message

Craig Markwardt wrote:
> Another thing that would be nice is warnings about ambiguous uses of
> variables that might shadow a function in the library. Note this
> code:
>
>   max = max(x, min=min)
>
> The next time you use min(x), will you get the function or the
> variable named min?
>
> You might say, "oh, I used square brackets, this is not a problem,"
> but since round parenthesis are still allowed (unless you disable
> them), I believe this can still be a problem. This has bitten me a
> *few* times, so I'm not saying it's a huge deal. It is a compile time
> issue since peoples' function libraries are different.

I believe a function takes precedence over an identically named array
subscripted with parentheses, e.g.

```
IDL> min = indgen(10)
IDL> a = indgen(5)
IDL> print, min(a)
```

```
     0
```

Using square brackets eliminates the ambiguity between the function name
and the variable name:

```
IDL> print, min[a]
     0      1      2      3      4
```

If you want to check whether a built-in or library function already
exists with a certain name (such as 'MIN'), first check the online help:

```
IDL> ? min
```

If it's not found there, perhaps there is a library function named
min.pro, and you could try compiling it:

```
IDL> .compile min
```

The only remaining possibility (I think) is that a function or procedure
with the same name is buried inside some other procedure or function yet
to be compiled. In any case, using square brackets for array
subscripting helps minimize name-space problems.

Cheers,
Liam.
http://cimss.ssec.wisc.edu/~gumley

Subject: Re: check for duplicate routine names?
Posted by Liam E. Gumley on Fri, 19 Jan 2001 16:55:41 GMT

View Forum Message <> Reply to Message

"Liam E. Gumley" wrote:
> If you want to check whether a built-in or library function already
> exists with a certain name (such as 'MIN'), first check the online help:

Of course, I do *not* recommend naming a variable 'MIN', even if it is
allowed in IDL. I just used this as an example. For minimum value
variables, I use a variable name like 'MIN_VALUE'.

Cheers,
Liam.
http://cimss.ssec.wisc.edu/~gumley

Subject: Re: check for duplicate routine names?
Posted by Jaco van Gorkom on Fri, 19 Jan 2001 17:06:49 GMT

View Forum Message <> Reply to Message

Pavel A. Romashkin wrote:

>
> If I had a vote, I'd be strongly against such checking.
> This means that any time I try to compile a routine which is already
> compiled, I'd be stopped by a warning message?! That would be a disaster
> for developing the code.

Warnings need not stop you. I figure IDL should be capable of printing a
two-line warning message without crashing or anything. And in most
cases, warnings are there for the programmer to ignore.
Apart from that, a warning would of course only be in order if the
compilation was done from a different source (be it .pro or .sav) then
the already compiled version.

cheers,
Jaco

Jaco van Gorkom gorkom@rijnh.nl
FOM-Instituut voor Plasmafysica Rijnhuizen

Subject: Re: check for duplicate routine names?
Posted by davidf on Fri, 19 Jan 2001 17:34:17 GMT

View Forum Message <> Reply to Message

Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:

> My take on it, avoid routines with generic names. It is tempting to
> write a program called "Just_Doit.pro", but it pays to add
> "prj1_just_doit.pro", unless you are writing a *library* function that
> will be used universally by your other projects.

Good advice here, and one I learned after being
knocked senseless more times than I care to admit.

Here is a rule for you and two corollaries.

The *last* program module in the file should
have the same name as the file. This is also
called the "command name" or "command module".

Any other program modules in the file will be
utility routines for the "command" module. They
will all have the command name pre-pended to
their actual names to remind you of their role

and to keep you out of a LOT of hot water.

If a utility routine needs to be used by
another command module, then that utility
routine has moved up in life (some will say
it has "transcended its stature"). Move it
into it's own command module file and put
it on your path.

This is know as the "Evolutionary Path of Program
Development". Follow this and life will be easy. :-)

Cheers,

David

P.S. Let's just say I haven't been able to put
down Ken Wilber's new update of his book "A
Brief History of Everything." What a wonderful
book!

--
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

Subject: Re: check for duplicate routine names?
Posted by Craig Markwardt on Fri, 19 Jan 2001 18:54:54 GMT

View Forum Message <> Reply to Message

"Liam E. Gumley" <Liam.Gumley@ssec.wisc.edu> writes:

> Craig Markwardt wrote:
>> Another thing that would be nice is warnings about ambiguous uses of
>> variables that might shadow a function in the library. Note this
>> code:
>>
>>   max = max(x, min=min)
>>
>> The next time you use min(x), will you get the function or the
>> variable named min?
>>
... deleted by Craig
>
> I believe a function takes precedence over an identically named array

> subscripted with parentheses, e.g.
>
> IDL> min = indgen(10)
> IDL> a = indgen(5)
> IDL> print, min(a)
>   0
>
> Using square brackets eliminates the ambiguity between the function name
> and the variable name:
...

Point taken. I was more trying to say, "because these things can
potentially be confusing, it would be nice to have a tool that
checks for them." My example of MIN/MAX was a little trivial; there
are possible clashes with, say, the Astronomy library which are not as
obvious.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: check for duplicate routine names?
Posted by Pavel A. Romashkin on Fri, 19 Jan 2001 21:34:26 GMT

View Forum Message <> Reply to Message

Aren't we tired yet of "Warning: Floating underflow" and such? If not, I
guess, another similar one will be allright.

Cheers,
Pavel

Jaco van Gorkom wrote:
>
> Pavel A. Romashkin wrote:
>>
>> If I had a vote, I'd be strongly against such checking.
>> This means that any time I try to compile a routine which is already
>> compiled, I'd be stopped by a warning message?! That would be a disaster
>> for developing the code.
>
> Warnings need not stop you. I figure IDL should be capable of printing a
> two-line warning message without crashing or anything. And in most
> cases, warnings are there for the programmer to ignore.

> Apart from that, a warning would of course only be in order if the
> compilation was done from a different source (be it .pro or .sav) then
> the already compiled version.
>
> cheers,
>   Jaco
>
> ------------------------------
> Jaco van Gorkom   gorkom@rijnh.nl
> FOM-Instituut voor Plasmafysica Rijnhuizen

Subject: Re: check for duplicate routine names?
Posted by Pavel A. Romashkin on Fri, 19 Jan 2001 21:40:27 GMT

View Forum Message <> Reply to Message

Paul van Delst wrote:

> It does seem of late that there are a number of new(er) posters who appear unfamiliar with the ways
> of the Coyote. :o)

Not to worry. If they hang around for a week or two, they will figure it
out: the Coyote likes to bite a little, but after that he appears to
feel so guilty and sorry for the victim that he provides them with
ready-to use code otherwise selling for big bucks :-)

Cheers,
Pavel

Subject: Re: check for duplicate routine names?
Posted by davidf on Fri, 19 Jan 2001 23:19:12 GMT

View Forum Message <> Reply to Message

Pavel A. Romashkin (pavel.romashkin@noaa.gov) writes:

> Not to worry. If they hang around for a week or two, they will figure it
> out: the Coyote likes to bite a little, but after that he appears to
> feel so guilty and sorry for the victim that he provides them with
> ready-to use code otherwise selling for big bucks :-)

Actually, I was sore because I had spent a whole weekend a
month or so ago putting my book on a CD-ROM in PDF format
for Michael and he never ordered it. :-(

Cheers,

David

P.S. But you're right. I'm a damn weeny.

--
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

Subject: Re: check for duplicate routine names?
Posted by R.Bauer on Mon, 22 Jan 2001 14:17:00 GMT

View Forum Message <> Reply to Message

Michael W Asten wrote:
>
> It is a stupid error to have a two routines of the same name
> DoitNow.pro , in two different library files MyLib1.pro and MyLib2.pro

Dear Michael!

Using a Library Routine (e.g. widgets normaly always having
al lot of routines in one file) some rules are necessary.

I. The library file

1.) The name of the library is the name of the last routine
2.) each other routine name begins whith the name defined by 1.)
    e.g. :
     x_synchronize is the calling routine name

     x_synchronize_event
     x_synchronize_sensitive
     x_synchronize_data
     ....
3.) The order of the routine is the order of needed.

II. The routine files

1.) Routines which are useful in several library files
    have to be standalone (not included to libraries)

2.) The names of the routine is the filename plus .pro

(lowercase)

III. Test of packages and distribute

1.) resolve_dependencies
This procedure will resolve dependencies of a given
IDL-procedure/function to the very end
This routine stops if a) a routine isn't found
b) a routine couldn't be compiled

For further routines and licensing please look first at
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml

http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_source/idl_
html/dbase/download/resolve_dependencies.tar.gz

2.) tar_gz_dependencies

This routine is used on a unix/linux system to make a tar.gz file of
all
routines used by one routine. RSI files are not included.

This procedure founds all dependent routines from a given routine.
They are saved into a tar.gz file
Additional Informations about the routines are saved into an html
file too.

regards

Reimar

--
Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-1)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de

===
a IDL library at ForschungsZentrum Jï¿½lich
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml

http://www.fz-juelich.de/zb/text/publikation/juel3786.html