
Subject: Re: Application Development

Posted by [Russell\[1\]](#) on Wed, 28 Oct 2015 19:32:22 GMT

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

Hi Heinz,

I'm working on this, but this still has the somewhat clunky requirement that I know a priori which objects and structure definitions are used a priori. I guess I could grep on "obj_new" at the command line, and then know which files use objects, but I think it'll be tougher to know which structures were used. But this is a good idea of a place to start.

thanks again,
Russell

On Tuesday, October 27, 2015 at 3:54:23 PM UTC-4, Heinz Stege wrote:

> On Mon, 26 Oct 2015 11:25:15 -0700 (PDT), @gmail.com wrote:

>

>> So is there a way to get a list of all the routines, including objects, for this project?

>

>

> Hello Russel,

>

> 24 hours and no answer. So I will tell you my solution(s), which may
> be a little bit exotic. [*]

>

> The simple way is to write an extra line of code for each object

> class. If FOO is the name of the object class, write

> if 0b then foo__define

> preferably in every routine creating objects of the class FOO. This

> statement may look a little bit strange, because this line is never

> executed. (0b of cause is always FALSE.) However the IDL routine

> ROUTINE_INFO,/UNRESOLVED responds to this line, and therefore

> RESOLVE_ALL compiles the file foo__define.pro. As you said, you have

> all methods of an object class in this file.

>

> The second way uses the same trick, but works with an extra file

> foo.pro:

>

> function foo,par1,par2,...,_ref_extra=extra

> return,obj_new('foo',par1,par2,...,_strict_extra=extra)

> foo__define

> end

>

> Again the line foo__define is never executed, however

> ROUTINE_INFO,/UNRESOLVED finds this procedure, when you write

> obj=foo(par1,par2,...,key1,key2,...)
> instead of
> obj=OBJ_NEW("foo",...)
> at object creation. (This is important!) Note that this way does not
> interfere with the "simplified object creation", introduced in IDL
> 8.0. Even better, it makes it possible for older IDL versions.
>
> I think, the second way would be beneficially, if you had separated
> files for your class methods, e.g. one file for every method. You
> would need to write the list of all the files at one place only.
>
> I hope, the above is clearly understandable.
>
> Cheers, Heinz
>
>
> [*] The best of cause would be, the IDL developers gave
> ROUTINE_INFO,/UNRESOLVED a new keyword to include the methods in the
> list of procedures/functions. However I don't know, if this is
> possible. ;-)
