

---

Subject: Re: auto-compile

Posted by [Robbie](#) on Fri, 21 Jul 2006 22:52:50 GMT

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

---

Ideally, we should be able to specify a compiler directive at the top of a .pro file to indicate that RESOLVE\_ROUTINE should compile the specified dependency.

In practice, I tend to arrange my code into modules which live in separate directories. I tend to compile directories rather than individual files. For example, you might have a few different algorithms which are invoked by CALL\_FUNCTION. I'd recommend keeping those in a module (subdirectory).

My trick for IDL library routines is to use UNIX soft links. I just create a directory which contains soft links to the IDL library routines of interest. I used to have a IDLgrObjects module directory for all graphics atoms, but I think that iTools resolve all takes care of that now.

Dunno what you'd do on windows..

Robbie

---

---

Subject: Re: auto-compile

Posted by [JD Smith](#) on Mon, 24 Jul 2006 16:55:56 GMT

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

---

On Fri, 21 Jul 2006 15:52:50 -0700, Robbie wrote:

- > Ideally, we should be able to specify a compiler directive at the top
- > of a .pro file to indicate that RESOLVE\_ROUTINE should compile the
- > specified dependency.
- >
- > In practice, I tend to arrange my code into modules which live in
- > separate directories. I tend to compile directories rather than
- > individual files. For example, you might have a few different
- > algorithms which are invoked by CALL\_FUNCTION. I'd recommend keeping
- > those in a module (subdirectory).

Those get taken care of in my library, it's when they occur elsewhere, including the IDL !DIR/lib library, that it's a problem.

- > My trick for IDL library routines is to use UNIX soft links. I just
- > create a directory which contains soft links to the IDL library
- > routines of interest. I used to have a IDLgrObjects module directory
- > for all graphics atoms, but I think that iTools resolve all takes care

> of that now.

And how do you discover which of these are being called indirectly via a CALL\_\*? Right now I use the "wait for it to break" method, which isn't ideal.

JD

---

---

Subject: Re: auto-compile

Posted by [greg michael](#) on Tue, 01 Aug 2006 09:52:46 GMT

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

---

I gradually accumulate a make file with a list of .compile xxx lines in it and a save,/routines at the end, which I copy and paste into the IDLDE to do the compilation. This also relies on the 'wait for it to break' method, but at least I know that any routine that's ever been called during development ends up there. I'd rather use a better method if anybody knows one...

Greg

---