
Subject: Re: IDL 8.0 compile_opt changes
Posted by [penteado](#) on Mon, 28 Dec 2009 03:44:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Dec 26, 7:21 pm, "H. Evans" <bloggs...@googlemail.com> wrote:

> So, what's the problem with using a "new" syntax that involves the
> comment character? This is what happens with numerous Unix shell
> scripts. For example:
>
> ;!compile_opt idl1
>
> which would not break old code, but would inform the new version that
> the old syntax should be used when interpreting the .pro file. This
> could/should be the first line of the file. This would make converting
> older files a doddle with the flexibility of setting the default
> compile_opt to idl2 while at the same time not breaking the file for
> older versions of IDL. Ideally a flexible syntax should be specified
> (could even be XML format if we really wanted to be absurd) that can
> be encapsulated in the commented header of a file.
>
> I'm still partial to a new file extension, though. As mentioned
> before, it worked quite well with the FORTRAN community (and let's
> face it, there hasn't been a compiled computer language to rival
> Fortran developed in the last 20 years.). And let's not forget that
> minor, but much needed upgrade to c: c++.

I used Fortran as an example because C++ is so far from C that it seems more like a new language rather than an upgrade. But same as Fortran, they also generally use the same compilers, which decide what to do based on the file extension.

An example of the trouble of a backward-incompatible update keeping the same extension is the current situation with Python. Python 3 has been out for about one year now, and still some major Linux distributions ship with Python 2, because of all the old software that would not work unchanged. Which slows down the transition, forcing the use of the older version in new software being developed now.
