
Subject: Re: IDL 8.0 compile_opt changes
Posted by [penteado](#) on Fri, 08 Jan 2010 13:36:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Jan 8, 7:29 am, "H. Evans" <bloggs...@googlemail.com> wrote:

- > - Anti-aliasing built into the direct graphics functions (or am I the
- > only person
- > left that finds the object graphics to be overly complex and tedious
- > to
- > produce a simple plot, while the direct graphics quality not of
- > publication
- > quality?) Alternatively, a complementary function for the old direct
- > graphics
- > routines that use object graphics, e.g. dgplot as a direct
- > replacement of
- > plot. Particularly one that can be used in batch mode (no attached X
- > or window
- > device - a problem with using the iTools for batch processing).

To me it seems that the iTools are designed to be near direct replacements of their direct graphics counterparts. It is possible to create iTools non-interactively and without an X server, just by using the keywords:

```
user_interface='none',/no_saveprompt,/disable_splash
```

Since IDL 7.1, there has been a much easier to use procedural interface to the iTools, which includes things like iopen, isave, and idelete. I always found that these new features should have been better advertised when they were introduced. See

<http://michaelgalloy.com/2009/08/15/idl-7-1-itools-procedural-api.html>
http://www.dfanning.com/itool_tips/invisible.html

For some examples.

- > - more functionality on accessing the contents of SAVE files (delete
- > variables, overwrite,
- > extract just a single named one, rename contents, etc.)

It is already possible to read select variables from save files, even select heap variables in them, which I have used to obtain only the targets of select elements in pointer arrays. This is done with the IDL_Savefile object. Though I would still like some more of the CMSVLIB functionality to be built-in.
