
Subject: Re: Looking for IDL code documentation standards
Posted by [Michael Wallace](#) on Mon, 16 May 2005 20:01:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

- > IDL coding style tends to be quite personal, even for RSI programmers.
- > Take a quick perusal through !DIR/lib and notice that they can't even
- > agree on whether to indent the body of routines!

Yeah, I've noticed that before. Lack of indentation drives me up the wall.

- > I whipped up this example coding standard, trivially enforceable by
- > IDLWAVE:
- >
- > http://idlwave.org/idlwave_template.pro.html
- >
- > Hope it is of some use.

Actually, except for the indentation rules, it is pretty close to the rules I made up for myself.

Reserved Words - lowercase (e.g. begin, end, do, for, if, while)

Variables: lowercase, words separated by "_" (e.g. my_var)

Routines: lowercase, words separated by "_" (e.g. my_func)

Keywords: UPPERCASE, words separated by "_" (e.g. MY_KEYWORD)

Classes: MixedCase, no separation character (e.g. MyWonderfulClass)

Methods: MixedCase, no separation character (e.g. GetProperty)

Indentation: 4 spaces for each block

Long lines: 8 spaces or align to previous line (programmer's discretion)

If...then...else is too long to fit on a single line, break into begin...end blocks rather than use line continuation. Same rule for while, for, etc. Simply put, use line continuation only when you have to.

Avoid multi-command lines (e.g. command1 & command2 & command3)

Use spaces around operators (e.g. var = foo + 2 ^ 32)

No spaces between function name and parenthesis (e.g. my_func(foo))

I've got several more spacing rules, a couple commenting rules and file naming conventions, but I don't want to spend the time writing them out.

And I'd probably wind up boring everyone to death. Either that or they'd just stop reading and go on to a thread with more excitement. :-)

-Mike
