
Subject: Re: Using `_extra` for all routines?
Posted by [meron](#) on Sat, 14 Oct 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <jackel.47.0D621ECF@canlon.physics.uwo.ca>, jackel@canlon.physics.uwo.ca (Brian Jackel) writes:

> In article <45ltaq\$8q7@llnews.ll.mit.edu> knight@ll.mit.edu writes:

>

>> Does anybody see any drawback to including `_extra` as a keyword in ALL
>> procedures and functions? My motivation is to allow keywords passed through to
>> multiple routines inside others. Here's an example:

>

>> Is this ok? One drawback might be ambiguity, two keywords whose names
>> conflict. Does anybody see any other drawbacks?

>

> One thing to watch out for is that any mis-spelled keywords will be absorbed
> by `_EXTRA`. No big deal if you're using `KEYWORD_SET()` to check on all legal
> keywords, and using lots of messages if they're not provided. However, if
> un-supplied keywords are being quietly given default values the user may never
> realize that they're using an improper keyword.

The biggest problem is that when a single routine calls few other, all the stuff in `_extra` will be passed down to the first one. If this one is a "bottom routine" and it doesn't recognize some of the keywords, it'll object. If it is not bottom, and is also defined with `_extra`, it'll pass the unidentified ones farther on, etc. Eventually chances are that you'll hit a "bottom" that doesn't recognize some of the keywords and it'll bomb. Of course you can deal with by pre and post processing (removing pieces from the structure defined by `_extra` before a call and returning them there after the call, but that's tedious and you have to know in advance what you may want to remove. A case like the one described by the previous poster (misspelled keyword) will slip right through.

`_extra` is a powerful stuff, but I recommend using it with care.

Mati Meron | "When you argue with a fool,
meron@cars3.uchicago.edu | chances are he is doing just the same"
