
Subject: Re: Passing optional parameters through a wrapper routine

Posted by [Craig Markwardt](#) on Wed, 09 Feb 2000 08:00:00 GMT

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

bowman@null.edu (Kenneth P. Bowman) writes:

```
> The _EXTRA keyword handles passing keyword parameters through a wrapper
> routine, but what about optional parameters? If my wrapper for TVRD has
> one of my own (mandatory) parameters, do I have to do the following?
>
> PRO TVRD_WRAPPER, my_arg, x0, y0, nx, ny, channel, _EXTRA = tvrd_keywords
>
> CASE N_PARAMS() OF
>   1 : image = TVRD(          _EXTRA = tvrd_keywords)
>   2 : image = TVRD(x0,      _EXTRA = tvrd_keywords)
>   3 : image = TVRD(x0, y0,   _EXTRA = tvrd_keywords)
>   4 : image = TVRD(x0, y0, nx, _EXTRA = tvrd_keywords)
>   5 : image = TVRD(x0, y0, nx, _EXTRA = tvrd_keywords)
>   6 : image = TVRD(x0, y0, nx, ny, channel, _EXTRA = tvrd_keywords)
> ELSE : MESSAGE, 'Incorrect number of arguments.'
> ENDCASE
```

I have a wrapper procedure in XFWINDOW.PRO which looks remarkably similar to this one. To counter David, I find that it's usually the cleanest and safest bet to mess with as *few* parameters and keywords as possible when wrapping another function. You can get into too much trouble trying to second-guess what the defaults should be.

The ugliness above could be solved if there would be a variable-parameter feature to the language. This might work the same way that _EXTRA passes keywords through. For example, in

```
PRO TVRD_WRAPPER, my_arg, $EXTRA_ARGS$, _EXTRA = tvrd_keywords
  image = TVRD($EXTRA_ARGS$,      _EXTRA = tvrd_keywords)
end
```

the black box variable \$EXTRA_ARGS\$ would somehow contain any additional arguments, and could be passed directly through to the wrapped function. The use of \$'s is syntactic sugar to be sure, but it's just an example.

Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
