Subject: Re: IDLVM and retall
Posted by David Fanning on Fri, 08 Sep 2006 04:00:24 GMT
View Forum Message <> Reply to Message

JD Smith writes:

> Here's a better example:
>
> pro MyClass:f1, ev
>   if something_is_wrong self->Error
>   do_calc
> end
>
> pro MyClass:f2,ev
>   self->f1,ev
>   do_another_calc
> end
>
>> Can this be a difference between Windows and UNIX?
>
> If you simply return (or return twice, if you were able to do that
> from within self->Error), you'd be right back inside MyClass:f2 and
> do_another_calc would run, which is not what you want. With RETALL,
> the entire calling stack is returned from, not just one step in the
> stack.

OK, I see what you mean. Yes, when you are trying to
merge widgets and objects things do tend to get a little
messy. I don't find myself buried this deep in my widget
programs, which is why RETURN has always worked for
me. But I do remember some messy, messy code in my
Catalyst Library to keep error messages from propagating
up the whole damn call stack! Essentially, we had to
keep track of whether the error had already been
"handled" or not, and keep issuing RETURNs if it had.
Your solution is probably better looking than this. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")