> No, there are a TON of errors in this program. You might just as well
> find all of them before you move on. :-)

I guess my point was to make a case WITH a ton of errors to create the cascade and see how CATCH works. I could check to see if "a" and "b" are defined (one of your earlier points) but I would still have to use RETURN in CALL_ADD instead of MESSAGE.

If I am 6 programs deep into the stack, I do not want to generate 6 error messages as my Catch blocks work their way back up. Also, putting RETURN instead of MESSAGE does not seem to be a general solution to errors. (... or is it?)

Maybe my definition of "utility function" is too strict and I should think about using ON_ERROR, 2 more.

Will keep pondering...