
Subject: Re: ERROR_MESSAGE

Posted by [Bob\[4\]](#) on Thu, 22 Sep 2011 23:41:20 GMT

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

I too have been burned by this and am curious if there is a nice solution. Sometimes I've had to kill the idl session to get out of it. For this reason and the fact that I almost always want to check some variables where the error occurred, I prefer using the default (on_error, 0) error handling to actually catching them. Catching the error is disorienting and make it harder to find the error. In big libraries (e.g. cg) it would be great if setting a system variable would switch from the catch method to the on_error method (probably using on_error, 2 since it is a library).

Subject: Re: ERROR_MESSAGE

Posted by [Allard de Wit](#) on Fri, 23 Sep 2011 07:57:38 GMT

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

Dear Wayne,

The solution is to put a RETALL or RETURN inside the error handler:

```
Catch, error_status
IF error_status NE 0 THEN BEGIN
  Catch, /CANCEL
  void = Error_Message()
  RETURN
ENDIF
```

This will get you out of the offending routine with all the pop-ups and still get you the traceback information that you want. Depending on what the program does with the error, you may even want to replace the RETURN statement with a "Message, /REISSUE_LAST" to propagate the error up to an error handler at a higher level.

Allard
