Subject: Re: Error Handling Advice
Posted by Matthew Argall on Mon, 14 Jan 2013 18:30:03 GMT
View Forum Message <> Reply to Message

Ok, I think I have reasoned with myself enough to know when to use ON_ERROR, 2 and when to use CATCH, but most CATCH blocks that I see just print an error message and return to the calling program, anyway, without doing anything special.

For example, take the following

```
----------%<----------
CATCH, Error_status

IF Error_status NE 0 THEN BEGIN
    CATCH, /CANCEL
    PRINT, 'Error index: ', Error_status
    PRINT, 'Error message: ', !ERROR_STATE.MSG
    RETURN
ENDIF
----------%<----------
```

Is there any reason for not using ON_ERROR, 2 in this case?

From what I gather, the CATCH blocks are normally used to make a program "fail gracefully", in that it resets the device and color table, closes files, frees pointers, etc. But without anything to reset, is ON_ERROR, 2 sufficient?

I guess I will start with some with trial and ::ehem:: error ;-)