

Update: SafeCall Defined

by *Brian Long*

Issue 51 contained an article by myself discussing the ins and outs of the Delphi `safeCall` directive. I foolishly thought I'd covered all the relevant points on the subject when the magazine went off to print.

However, I recently received an email from Neil Poulton that put me right on one point. In the article, I made great play of the fact that `safeCall` takes care of the `HResult` return code and does not let you gain access to it. This turns out to be only half right.

If you wish to return a custom `HResult` that represents an error (which means has the high bit set) you can. All you need to do is raise an `E01eSysError` exception. However, custom `HResult` values that

represent successful operation cannot be specified this way.

The constructor for `E01eSysError` takes three parameters which are used by `HandleSafeCallException` (in the `ComObj` unit) to fill in the details of the COM error object. The first parameter is a message: either a custom message or, if the `HResult` is a standard Windows `HResult`, an empty string. The exception class will ask Windows for a string describing the `HResult`. The second parameter is the `HResult` value itself and the last is a help context number.

The implementation of `HandleSafeCallException` has a small test that I missed when researching the article. Assuming the raised exception in the `safeCall` method is an

`E01eSysError` exception, and the error code that it holds is less than 0, then that error code will be returned as the `HResult` from the `safeCall` method instead of `E_UNEXPECTED`.

The test for the error code being less than 0 is present in Delphi 3, 4 and 5. Now it is quite well known that Delphi 4 redefined `HResult` to be an unsigned number (which cannot hold negative numbers). So there is a *potential* for the test to be insufficient to catch error `HResult` values in Delphi 4. However, the `E01eSysError` `ErrorCode` property is fortunately defined as an `Integer` in Delphi 3 and 4. Delphi 5 redefines it to be an `HResult`. This means that checking for values less than 0 will correctly pick up error `HResult` values (those with their high bit set).