

#4: Error Returns from GetNewDialog

See also: The Dialog Manager

Written by: Russ Daniels

April 4, 1985

Updated:

March 1, 1988

When calling `GetNewDialog` to retrieve a dialog template from a previously opened resource file, how are error conditions indicated to the caller?

Unfortunately, they aren't. The Dialog Manager calls `GetResource` and assumes the returned value is good. Since the Dialog Manager doesn't check, you have two choices. Your first choice is to call `GetResource` for the dialog template, item list, and any resources needed by items in the item list yourself. But what do you do when you find the resources aren't there? Try to display an alert telling the user your application has been mortally wounded? What if resources needed for the alert aren't available?

The second, simpler alternative is to assure that the dialog template and other resources will be available when you build your product. This is really an adequate solution: If somebody uses a resource editor to remove your dialog template, you can hardly be blamed for its not executing properly.

A good debugging technique to catch this sort of problem is to put the value `$50FFC001` at absolute memory location 0 (the first long word of memory). If you do that, when the Dialog Manager tries to dereference the nil handle returned by the Resource Manager, you'll get an address error or bus error with some register containing `$50FFC001`. If you list the instructions around the program counter, you'll often see something like:

```
MOVE.L (A2),A1      ; in effect (0),A1
MOVE.L (A1),A1      ; the error occurs here
```

`GetNewWindow` and most of the other "GetSomething" calls will return nil if the "something" is not found.