Subject: Close() callback not conditioned in a dialog
Posted by forlano on Tue, 07 Nov 2006 20:28:18 GMT

Hello,

in some dialog containing for example an edit widget I perform a control to avoid if it is empty. So I force the flag Notnull in the designer for that widget. The dialog has then an OK and Close buttons.
The problem: when the user press Close o whatever button that performs a Close() action the dialog does not close if the user do not enter some weird thing in the edit widget. So the widget has priority on the close callback.
I think that the closure of the dialog connected with the Close() callback should be immediate and irrespective of the condition of the others widgets (not null for example), because these eventual data will lost be in any case. It should be responsability of the programmer eventually confirm the closure of the dialog.
This is particularly annoying if the dialog contains many ctrls with not null flag activated.
Of course this problem can be bypassed removing the Notnull flag and performing a control on each widget but this seems to me not natural if the notnull method exists.

Luigi

Subject: Re: Close() callback not conditioned in a dialog
Posted by mirek on Thu, 09 Nov 2006 19:33:47 GMT

Then make your Close button a Rejector (or simply Breaker)!

There are three modal loop breakers:

- Breaker simply exits it
- Acceptor accepts widgets first (accepting makes the check for Null)
- Rejector rejects all widgets (this can be different from simple Break sometimes, e.g. open database data are canceled etc..)

Mirek

Subject: Re: Close() callback not conditioned in a dialog
Posted by forlano on Thu, 09 Nov 2006 20:48:54 GMT

luzr wrote on Thu, 09 November 2006 20:33Then make your Close button a Rejector (or simply Breaker)!

There are three modal loop breakers:

- Breaker simply exits it
- Acceptor accepts widgets first (accepting makes the check for Null)
- Rejector rejects all widgets (this can be different from simple Break sometimes, e.g. open database data are canceled etc..)

Mirek

I've used the following code line


```
//btnClose.WhenAction = THISBACK(Close); // previous
btnClose <<= Breaker(999); // new
```


and it works as I wanted. Now the Breaker is promoted to my default method for Close/Exit button.

Thanks!
Luigi