Subject: Dialog boxes : why struc instead of class ?
Posted by jibe on Fri, 18 Mar 2011 09:04:43 GMT
View Forum Message <> Reply to Message

Hi,

In most examples, a dialog box is done this way :
struct MyDlg : public WithMydialogLayout<TopWindow> {
...
}
Why not use a class, something like this :
class MyDlg : public WithMydialogLayout<TopWindow> {
public:
...
protected:
...
private:
...
}

This looks surprising for me, more especially when I see
typedef MyStruc CLASSNAME;

The best reason is surely that it's working well this way, but it's very confusing for beginners !

Subject: Re: Dialog boxes : why struc instead of class ?
Posted by dolik.rce on Fri, 18 Mar 2011 12:29:42 GMT
View Forum Message <> Reply to Message

Hi Jibe,

In C++, the only difference between struct and class are in default access right to the members. Class has private members as defualt, struct uses public, otherwise the behavior is the same. So using struct for dialogs in U++ is just laziness to write class XYZ{ public: ... } You can of course use whatever form you like better, there is no functional difference.

The CLASSNAME could be equally well STRUCTNAME (or ASFDGFSDGSDFG or whatever else ). I guess it kind of makes sense to call it CLASSNAME though, because class is well defined, while struct can be C struct or C++ struct and that could sound a bit ambiguous (not sure how much difference there is if any...).

Best regards,
Honza

Subject: Re: Dialog boxes : why struc instead of class ?

Posted by jibe on Fri, 18 Mar 2011 14:01:16 GMT
View Forum Message <> Reply to Message

Ok, so nothing extraordinary !

It's just confusing for C++ developpers who learned to use class instead of struc, to be able to make an overridden class if necessary... But it's true that it's not very useful with the dialog boxes of an application !

Thanks for this explanation

Subject: Re: Dialog boxes : why struc instead of class ?
Posted by mirek on Fri, 18 Mar 2011 14:16:19 GMT
View Forum Message <> Reply to Message

jibe wrote on Fri, 18 March 2011 10:01
It's just confusing for C++ developpers who learned to use class instead of struc, to be able to make an overridden class if necessary... But it's true that it's not very useful with the dialog boxes of an application !

Actually, I am using mostly struct for dialogs in my "end-user" apps.

I guess that at the top layer of code, access protection is not that important.

Subject: Re: Dialog boxes : why struc instead of class ?
Posted by jibe on Fri, 18 Mar 2011 14:26:00 GMT
View Forum Message <> Reply to Message

Its true that it's not important at the top layer of code. Just used to do always the same way... It's what we learned when studying C++, but it's just theory !