Subject: General Menu Construction
Posted by Werner on Sat, 13 May 2006 10:52:45 GMT
View Forum Message <> Reply to Message

So far I have found 5 different menu construction patterns. Reducing the code to the crucial parts and unifying it yields:

#1 The MenuSet variant
(Used in: Bombs, CodeMetric, Puzzle, Scribble2, Scribble3, UWord, GUI Tutorial Menu, GUI Tutorial Adding Images, GUI Tutorial Toolbar, GUI Tutorial StatusBar And Decorative Frames, Menu)
• MenuBar menu;
• AddFrame(menu);
• menu.Set(THISBACK(A_Function));
• bar.Add("a_name", THISBACK(Another_Function));

#2 The MenuAdd variant
(Used in: HelloWorld)
• MenuBar menu;
• AddFrame(menu);
• menu.Add("a_name", THISBACK(A_Function));
• bar.Add("a_name", THISBACK(Another_Function));

#3 The ContextMenu variant
(Used in: GUI Tutorial Context Menu)
• MenuBar menu;
• MenuBar::Execute(THISBACK(A_Function));
• bar.Add("a_name", THISBACK(Another_Function));

#4 The ContextMenu Alternative variant
(Used in: GUI Tutorial Context Menu Alternative And Using CallbackArgTarget, Some More Stuff About Callbacks)
• MenuBar menu;
• CallbackArgTarget<int> result;
• menu.Add(AsString(i), result[i]);
• menu.Execute();

#5 The Layout variant
(Used in: AddressBook, AddressBookXML)
• LAYOUT(AddressBookLayout, width, height) // MenuBar var;
• ITEM(MenuBar, menu, Position()) // parent.Add(layout.menu;)
• menu.Set(THISBACK(A_Function));
• bar.Add("a_name", THISBACK(Another_Function));

Are there even more patterns?

When should I use “menu.Set” (#1) and when “menu.Add” (#2)?

Werner

Subject: Re: General Menu Construction
Posted by mirek on Sat, 13 May 2006 11:46:07 GMT
View Forum Message <> Reply to Message

Always use #1 instead of #2. Actually, we are investigating how to avoid #2 at interface level (which is OTOH quite difficult while preserving context menu system). The fact that it was left in HelloWorld is rather bug than anything else (now fixed).

The thing is that "Set" and "PopUp" do some additional actions to get menu ready (like automatic hotkey assignment).

As for "Layout" variant, it is in fact the same as #1 - it does not matter whether you use menu as frame or as view ctrl.

Also, the difference between both context variants is zero - all the difference is in CallbackArgTarget, not in MenuBar.

Mirek

Subject: Re: General Menu Construction
Posted by Werner on Sat, 13 May 2006 12:11:22 GMT
View Forum Message <> Reply to Message

Thanks for your instantaneous answer.

lay.h is a very nice piece of software, by the way.

Werner

Subject: Re: General Menu Construction
Posted by Mark999 on Sat, 28 Jul 2007 22:20:09 GMT
View Forum Message <> Reply to Message

I was wondering the same thing. I think you add from top down. For example:

```
CtrlLayout(*this, "This App");
AddFrame(menu);
AddFrame(TopSeparatorFrame());
AddFrame(toolbar);
AddFrame(statusbar);
```

```
//  add to each frame
        menubar.Set(THISBACK(MainMenu));
        .
        .
```

Is this correct?

Subject: Re: General Menu Construction
Posted by mirek on Wed, 01 Aug 2007 16:11:24 GMT
View Forum Message <> Reply to Message

Well, if the question is about Frame(s), the correct answer is perhaps "from outside to inside": each frame in order reduces the view size. E.g. menu normally occupies top part of window, scrollbar right etc...
