Subject: Re: Basic question regarding callbacks
Posted by cbpporter on Mon, 21 Jun 2010 12:12:13 GMT
View Forum Message <> Reply to Message

Q1: To understand what is happening there you need to understand function pointers and delegates. If you have a function pointer, you can then call that function through the pointer and transfer execution to an arbitrary piece of code. But in OOP, you can't have a pointer to a method of the object. Calling the method has no meaning if you don't know on what object to call it. This is where delegates come in. Basically, a delegate is an ordered pair formed from an object and a method of it that is going to get called. This is a very superficial explanation, but I hope that if will get you started.

THISBACK is just a shortcut to create a delegate where the object is "this".

So when you are using the Set method, that method will call the method MainMenu of your MyAppWindow instance (your current this) and it will pass it a Bar object, which in you case is the "menu" variable. In MainMenu method, you add the top level menus to you menubar, and pass them additional callbacks which get called for each item. This seems confusing at first because you are using the same mechanism both to construct menus (structure) and to define behavior. The difference is only in you intention, there are not separate mechanisms.

Q2: Since you are dealing with callbacks/delegates, not function pointers, it is normal to have one extra argument there for the object which has the method. So if first argument is the object, and the second is your intended argument, you get a Callback1. And you are not passing that argument manually because Set does that.

Q3: The argument bar is passed by the Set method and in your case it is "menu", an instance of MenuBar which is in the hierarchy that allows it to be passed as an argument to that function. The type is Bar as a common ancestor so you can use the same methods to construct both menubars and toolbars (and theoretically any kind of controls that are compatible with a Bar: one could imagine a TabCtrl with the same interface or an "Outlook" kind o pane control).

Q4: I don't know exactly where you get the loop, but it is normal to eventually enter an infinite loop. This is how GUI application work. They wait in a loop for user or system messages.