Subject: bug: file remaining in last-used list after Ctrl+W vs close-by-mouse
Posted by mr_ped on Wed, 01 Feb 2023 08:21:06 GMT
View Forum Message <> Reply to Message

The issue is that when File/Close file tab, or Ctrl+W is used, the code path is (WhenClose callback is NOT invoked):
idebar.cpp

```
  menu.Add(AK_CLOSETAB, THISBACK(ClearTab))
      .Help("Close the current file tab");
```

EditorTabBar.cpp

```
void Ide::ClearTab()
{
 int c = tabs.GetCursor();
 if(c >= 0)
  tabs.Close(c);
}
```

When mouse is used to press the cross, it hits the code path (does invoke WhenClose callback):
TabBar.cpp

```
void TabBar::LeftDown(Point p, dword keyflags)
...
   int ix = cross;
   if (!CancelClose(v) && !CancelCloseSome(vv) && ConfirmClose(v)) {
    WhenClose(v);
    WhenCloseSome(vv);
    TabClosed(v);
    Close(ix);
   }
```

The tab-context selecting "Close" hits code path (does invoke WhenClose callback):
TabBar.cpp

```
void TabBar::ContextMenu(Bar& bar)
...
  bar.Add(tabs.GetCount() > mintabcount, t_("Close"), [=] {
   if (!CancelClose(tabs[ii].key)) {
    WhenClose(tabs[ii].key);
    TabClosed(tabs[ii].key);
    tabs.Remove(ii);
    MakeGroups();
    Repos();
    SetCursor(-1);
```

```
  }
});
```

I was considering to fix this myself, but I'm not certain what is correct fix here, this looks wrong design to me, IMHO the WhenClose should be rather part of the void TabBar::Close(int n, bool action), or having one more extra wrapper like CloseWithEvents, but that means it would change order of the other calls.

Which is another confusing point, each close-tab path looks considerably different, the context menu is using tabs.Remove and does some groups/repos/cursor stuff, left click on cross doesn't do any of that, and file/close tab (Ctrl+W) is even more minimal. This looks suspicious to me.

@Mirek: any advice how to fix to your liking, or you want to refactor it yourself?
