Subject: Re: ArrayCtrl: GPF when thread Add(), PopUpEx, and Scroll collide
Posted by bushman on Sat, 06 Jul 2013 15:39:33 GMT
View Forum Message <> Reply to Message

Tks for the prompt response.

Quote:The root of problem is stupid M$ decision that binds windows and event loops to threads. That makes practically only possible to create windows and run event loops in the main thread.

I thought it was something rather OS-independent, since the issue apparently occurs under Linux too, according to Alendar's posting in this topic. Anyway, I wonder, would it be worth to ask how M$ does it? In other words, how a pure M$ app is supposed to be developed so that it does what I want without crashing? I mean, is there some M$ recipe for creating new windows not in main thread and still be able to run event loops in it without crashing? I don't know, maybe such an investigation could bring up new insights on which course to take to fix the problem, for I guess purely M$-based progs must end up having to do it one way or another.

Quote:Alternatively, I am starting to thing that perhaps easiest is to ban creation of windows in non-main thread (Prompts could perhaps be supported as exception, DisplayPopup::Sync would have to be rewritten).

If you decide to do so, does it mean we will no longer be able do stuff like adding Ctrls in a background thread dynamically, while 'simultaneously' enabling user interaction with GUI, or would new DisplayPopup::Sync code otherwise still make it possible?

tks.