Subject: Re: Defining a callback that respects timers?
Posted by dg86 on Wed, 16 Sep 2015 12:46:46 GMT
View Forum Message <> Reply to Message

On Wednesday, September 16, 2015 at 5:46:31 AM UTC-4, superchromix wrote:
> I don't know the answer to this question, but I am very interested in the solution, if one exists.
>
> I also write user interfaces which need to periodically update, while another process (e.g. data analysis) is ongoing.
>
> Mark

One possibility might be to create a superclass of the timer object that stores its anticipated firing time as a property that can be polled with GetProperty. Any widget callback that might block the timer then could check to see if it's time to fire and, if necessary, could (1) cancel the timer, and (2) call the timer's callback.

I could imaging storing the deadline time as part of the userdata in the timer.set() method.

Here's the catch: Timer() is defined as a set of built-in static methods (no source code, I believe). So, I also don't see how to implement my own static Timer method to inspect the userdata.

It would be _REALLY_ nice to be able to set a timer that preempts widget callbacks (pre-8.4 behavior), perhaps as an option to the Timer.Set() method.