
Subject: Re: Concurrent widget program.
Posted by [Andy Loughe](#) on Tue, 02 Jun 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

- > But that is not the case at all. Consider this example.
- > Suppose the event handler code went into a processing loop
- > and that the loop took 5 minutes to execute. Then the user
- > could be pushing as many buttons on as many widget programs
- > as he or she liked, but nothing would be happening. In
- > fact, nothing at all would happen until that loop finished
- > and then, probably, all hell would beak loose as IDL rushed
- > to handle all the events that had queued over the past five
- > minutes.
- >
- > IDL does only one "thing" at a time. It is not a multi-threaded
- > or multi-tasking program. This is exactly why you try to
- > avoid writing loops in event handler code. In fact, when you
- > need a loop, you try to take advantage of the widget program
- > itself *acting* like a loop. A widget animation is a perfect
- > example of this. If a widget animation was really in a loop,
- > there would be no way to interrupt the animation with, for
- > example, a Quit button. You can look at the example XMOVIE on
- > my web page for an example of how to use the program itself
- > to simulate a loop. You will see that each "event" actually
- > displays just a single frame of the animation sequence.
- > Between one event and the next a Quit button event, for
- > example, could be queued up and processed.

I am not a widget genius, but isn't that one reason you add the hourglass keyword when controlling widgets that take "awhile" to process the event?

--
Andrew F. Loughe |
afl@cdc.noaa.gov
University of Colorado, CIRES Box 449 |
http://cdc.noaa.gov/~afl
Boulder, CO 80309-0449 | phn:(303)492-0707
fax:(303)497-7013

"I do not feel obliged to believe that the same God who has endowed us with sense, reason, and intellect has intended us to forego their use."
-Galileo