

Control Structure blockReentry Field

Long ago, on Macintosh version 6 and before, if a control action procedure took a long time to run, it was possible to repeat the action and start the action procedure again before the previous invocation was finished (that's called reentrancy). That is, if a button starts a long computation, like a lengthy curve fit, it was possible to start a new fit before the first was done, if you clicked the button again too soon. To control that, the blockReentry member was added.

In Igor 7 and later, code was added that prevents reentry, so the blockReentry field is obsolete. It is retained for backward compatibility and has no effect.

If you find a case in which reentry is a problem, please contact support@wavemetrics.com with a report. We will need an example that we can use to reproduce the problem.

Control Structure blockReentry Advanced Example

This example further illustrates the use of the blockReentry field. It is of interest only to those who want to experiment with this issue.

The ReentryDemoPanel procedure below creates a panel with two buttons. Each button prints a message in the history area when the action procedure receives the "mouse up" message, then pauses for two seconds, and then prints another message in the history before returning. The pause is a stand-in for a procedure that takes a long time.

The top button does not block reentry so, if you click it twice in quick succession, the action procedure is reentered and you get nested messages in the history area.

The bottom button does block reentry so, if you click it twice in quick succession, the action procedure is not reentered.

Because of architectural differences, reentry of the action procedure occurs on Macintosh only in Igor6 and does not occur at all in Igor7 or later. Because reentry may be affected by internal changes in Igor or by operating system changes, it may reappear as an issue in the future.

```
Function ButtonProc(ba) : ButtonControl
    STRUCT WMBUTTONACTION &ba

    switch( ba.eventCode )
        case 2: // mouse up
            // Block bottom button only
            ba.blockReentry= CmpStr(ba.ctrlName,"Block") == 0
            print "Start button ",ba.ctrlName
            Variable t0= ticks
            do
                DoUpdate
                while(ticks < (t0+120) )
                    Print "Finish button",ba.ctrlName
                break
            endswitch

    return 0
End

Window ReentryDemoPanel() : Panel
    PauseUpdate; Silent 1// building window...
    NewPanel /K=1 /W=(322,55,622,255)
    Button NoBlock,pos={25,10},size={150,20},proc=ButtonProc,title="No Block Reentry"
    Button Block,pos={25,50},size={150,20},proc=ButtonProc,title="Block
Reentry"
End
```