

CSplitterWnd and OnSize

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2004-07/1716.html>

From: Michael Chisholm (*chisholm_at_mitre.org*)

Date: 07/22/04

Date: Thu, 22 Jul 2004 15:06:11 -0400

I've seen people complain about the following problem in many posts. Others all seem to answer with the same solution—which does not work. Perhaps it worked in an earlier version of the libraries... I don't know. But I still have the problem, and would like a (nice) workaround.

I'm sure I can work around it in an ugly way, but I would like to know the "right" way to do this. I have Visual Studio .NET 2003 (Microsoft Visual C++ .NET 69462-270-0000007-18596)

My main frame uses a CSplitterWnd in its client area. I want to control how the panes resize when the user resizes the main frame. But CMainFrame::OnSize() is being called before the CSplitterWnd is completely created, so it is in an inconsistent state, and the operations I need to do on it cause the program to crash (NULL pointers get dereferenced). I need a way to determine whether the splitter is "ready" for me to do stuff with it. The only "solution" I've seen posted is to call CSplitterWnd::GetSafeHwnd() and compare to NULL. But that method returns a non-NULL value even when the splitter is not ready, so it does not work. FWIW, the code is below (the crash is in the first GetRowInfo() call). Any help would be appreciated.

```
BOOL CMainFrame::OnCreateClient(LPCREATESTRUCT lpcs, CCreateContext*
pContext)
{
    // TODO: Add your specialized code here and/or call the base class
    // return CFrameWnd::OnCreateClient(lpcs, pContext);

    if (!splitter.CreateStatic(this, 2, 1))
        return FALSE;

    ...etc...
}

void CMainFrame::OnSize(UINT nType, int cx, int cy)
{
    CFrameWnd::OnSize(nType, cx, cy);

    // TODO: Add your message handler code here
    if (splitter.GetSafeHwnd() != NULL)
```

```
{
  CRect r;
  GetClientRect(&r);

  int upperRowHeight, utemp;
  int lowerRowHeight, ltemp;
  splitter.GetRowInfo(0, upperRowHeight, utemp);
  splitter.GetRowInfo(1, lowerRowHeight, ltemp);

  //preserve the height of the lower pane if possible
  upperRowHeight = r.bottom - lowerRowHeight;

  splitter.SetRowInfo(0, upperRowHeight, utemp);
  splitter.RecalcLayout();
}
```

Andy