Subject: TimerCallBack interval resolution
Posted by deep on Wed, 19 Jan 2022 06:41:00 GMT
View Forum Message <> Reply to Message

Hi

I am experimenting with TimerCallBack. I want to set callback interval to say 25 ms.
To check this I have done small code sample. It is attached to this message.
I am measuring delay with GetTickCount(). Running this for 10 sec. And counting - Histogram.

My observation shows time aligns to 20 ms

Timer Delay -20
Time Count
20 320
21 160
22 5
23 2
28 1
37 1

Timer Delay -25
Time Count
28 1
30 1
34 1
40 102
41 142

Timer Delay -19
Time Count
19 2
20 331
21 147
22 5
23 1
24 1
31 1
60 1

Timer Delay -22
Time Count
22 3
23 2
24 5

```
40   116
41   113
43   1

Timer Delay -38
Time   Count
40   114
41   131
55   1


#include <CtrlLib/CtrlLib.h>

using namespace Upp;

struct TimerCallBackVariation : TopWindow
{

 Vector<int> interval;
 dword prev, now;

 TimeCallback timer, process;
 int timerDelay, processDelay;
 LineEdit le1;
 String Result;

 void Timer()
 {
  now = GetTickCount();
  interval.Add ( now - prev );
  prev = now ;
 }


 void Process()
 {
  timer.Kill();

  Result << "Count - " << interval.GetCount() << "\n";
  Result << "Interval Vector - " << interval << "\n" ;

  Sort ( interval );

  Result << "Sorted - " << interval << "\n";

  VectorMap<int, int> hist;
  int j, k;
```

```
  for ( int i = 0; i < interval.GetCount(); i++ )
  {
   j = interval[i];
   k = hist.Find ( j );

   if ( k > -1 )
   {
    hist[k] = hist[k] + 1;
   }

   else
   {
    hist.Add ( j, 1 );
   }
  }

  Result << "\nTimer Delay " << timerDelay << "\n";

  Result << "Time  \tCount" "\n";

  for ( int i = 0; i < hist.GetCount(); i++ )
  {
   Result << hist.GetKey ( i ) << "  \t" << hist.GetValues() [i] << "\n";
  }

  this->Title ( "Done" );
  le1.Paste(Result.ToWString());
 }

 TimerCallBackVariation()
 {
  timerDelay = -38;
  processDelay = 10000;
  Title ( "Timer Callback Time Variation" ).Zoomable().Sizeable();
  Add(le1);
  le1.SizePos();
  prev = GetTickCount();
  timer.Set ( timerDelay, [=] {Timer();} );
  process.Set ( processDelay, [=] { Process();} );
 }

};

GUI_APP_MAIN
{
 TimerCallBackVariation app;
 app.Run();
}
```

File Attachments

1) TimerCallCheck.tar.bz2, downloaded 300 times