
Subject: Time callbacks when ticks overlap
Posted by [Zbych](#) on Fri, 23 Jul 2010 09:03:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I have a problem with time callbacks. They use `GetTickCount` which overlaps every ~50 days and do all calculations on 32-bit variables. Unfortunately `Ctrl::TimerProc` doesn't work correctly when `(GetTickCount + delay)` overlaps.

For example when `GetTickCount` returns `0xFFFF0000` and I want to set time callback to 2 minutes (`0x1D4C0` ms), `TimeEvent.time` is set to `0x0000D4C0` and `Ctrl::TimerProc` executes callback every time (because this statement "`list->GetNext()->time < time`" is true). This situation is more likely to happen on posix, because `GetTickCount` returns ticks since the Epoch, not system power on.

My proposition is to change time comparison:

old:

```
while(list->GetNext() != list && list->GetNext()->time < time) {
```

new:

```
while(list->GetNext() != list && ((int)(time - list->GetNext()->time)) > 0) {
```

Casting to a signed number is a must.

and to remove this (I think it is unnecessary):

```
if(e->time > 0x80000000)  
    e->time = 0;
```

Subject: Re: Time callbacks when ticks overlap
Posted by [mirek](#) on Fri, 23 Jul 2010 09:11:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Zbych wrote on Fri, 23 July 2010 05:03Hi,

I have a problem with time callbacks. They use `GetTickCount` which overlaps every ~50 days and do all calculations on 32-bit variables. Unfortunately `Ctrl::TimerProc` doesn't work correctly when `(GetTickCount + delay)` overlaps.

For example when `GetTickCount` returns `0xFFFF0000` and I want to set time callback to 2 minutes (`0x1D4C0` ms), `TimeEvent.time` is set to `0x0000D4C0` and `Ctrl::TimerProc` executes callback every time (because this statement "`list->GetNext()->time < time`" is true). This situation is more likely to happen on posix, because `GetTickCount` returns ticks since the

Epoch, not system power on.

My proposition is to change time comparison:

old:

```
while(list->GetNext() != list && list->GetNext()->time < time) {
```

new:

```
while(list->GetNext() != list && ((int)(time - list->GetNext()->time)) > 0) {
```

Casting to a signed number is a must.

and to remove this (I think it is unnecessary):

```
if(e->time > 0x80000000)
    e->time = 0;
```

Thank you. You are right, the code was flawed, yours is correct.

Just to be sure:

```
void Ctrl::TimerProc(dword time)
{
    if(IsPanicMode())
        return;
    sTimerLock.Enter();
    TimeEvent *list = tevents();
    sTClick = time;
    sTimerLock.Leave();
    Ctrl::CheckMouseCtrl();
    Ctrl::SyncCaret();
    sTimerLock.Enter();
    while(list->GetNext() != list && ((int)(time - list->GetNext()->time)) > 0) {
        TimeEvent *e = list->GetNext();
        e->Unlink();
        if(e->delay < 0)
            sTimeCallback(time - e->delay, e->delay, e->cb, e->id);
        eventid++;
        sTimerLock.Leave();
        e->cb();
        sTimerLock.Enter();
        delete e;
    }
    sTimerLock.Leave();
}
```

Subject: Re: Time callbacks when ticks overlap
Posted by [Zbych](#) on Fri, 23 Jul 2010 12:01:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Fri, 23 July 2010 11:11
Just to be sure:
[cut]

That is ok. But there is one more place where time is compared:

```
static void sTimeCallback(dword time, int delay, Callback cb, void *id) {  
[...]  
for(e = list->GetNext(); e != list && ((int)(time - e->time) >= 0); e = e->GetNext());  
[...]
```

Subject: Re: Time callbacks when ticks overlap
Posted by [mirek](#) on Sun, 25 Jul 2010 07:31:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

Zbych wrote on Fri, 23 July 2010 08:01luzr wrote on Fri, 23 July 2010 11:11
Just to be sure:
[cut]

That is ok. But there is one more place where time is compared:

```
static void sTimeCallback(dword time, int delay, Callback cb, void *id) {  
[...]  
for(e = list->GetNext(); e != list && ((int)(time - e->time) >= 0); e = e->GetNext());  
[...]
```

Ops, thanks, missed that one...

That is why I prefer full zipped files posted as patch

Mirek
