Subject: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Fri, 18 Dec 2015 13:58:24 GMT
View Forum Message <> Reply to Message

Hi!

I have this program that runs other tasks using LocalProcess. The execution can take sometime a lot of time, and can even hang, so I decided to try MT. I used the callback posting mechanism to post a callback to the application with a new output fragment. And I'm using a LineEdit to show the output as it is created.

I saw no obvious way of inserting something to the end of the text in the LineEdit, so I used edit.Set(edit.Get() + t). This was really slow and unresponsive. Now I'm trying to use Insert and some sort of a position returned by LineEdit. But the position system that this control uses isn't the most intuitive and I always had to use a lot of trial and error to get it right. Using Append now is much more responsive (still not enough, I probably need to update it only every 1K of output or something), but the output is mangled, which makes me believe that I am using a wrong position as a parameter to Insert.

So my question is: how to easily and efficiently append something to LineEdit in a MT environment. And should I use GUI lock instead of post-backs?

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Fri, 18 Dec 2015 14:34:07 GMT
View Forum Message <> Reply to Message

I did some further testing. I am using a task that outputs all numbers form 0 to 99999 as a test. Running it the Windows console is very slow, over 4 seconds. But running it though LocalProcess is about 230 ms. So my goal is to get relatively close to 230 ms (300-400 ms will do) while still having the program be responsive.

I tested:
1. Updating the whole LineEdit. Unusably slow of course.
2. Using Insert and then ScrollEnd. Ignoring the corrupted output, I got almost 3 seconds.
3. Using only Insert. 2.4 seconds.
4. Using my favorite control: ColumnList! Plus using SetCursor to move to last element. I need to pre-process the output since ColumnList does not understand '\n', but it only takes 280 ms.
5. ColumnList and no SetCursor: somehow on average 10 ms slower :).

So the conclusion is that maybe LineEdit is not the right control to use here?

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Fri, 18 Dec 2015 18:24:24 GMT

All this sound very similar to theide Console... ide/Console.cpp :)

And yes, best it to append by insert at the end. Use GuiLock.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Mon, 21 Dec 2015 11:39:55 GMT

Yeah, using the code from Console that manipulates selection and uses Paste I got 11 seconds.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Didier on Mon, 21 Dec 2015 18:15:21 GMT

Hi cbporter,

I may be wrong, but It looks like you are trying to display every single update while you also want quick display.
Maybe you should disconnect display frequency from update frequency by appending all text adds in between two displays in one big text update (if this is possible with you're needs of corse)

I did something close to this when managing undo stack in my GraphCtrl : when doing quick zooming or panning (less than 300ms in between two actions), I append all actions in order to make one final undo action.

Hope this helps

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Tue, 22 Dec 2015 13:03:35 GMT

Didier wrote on Mon, 21 December 2015 20:15Hi cbporter,

I may be wrong, but It looks like you are trying to display every single update while you also want quick display.
Maybe you should disconnect display frequency from update frequency by appending all text adds in between two displays in one big text update (if this is possible with you're needs of corse)

I did something close to this when managing undo stack in my GraphCtrl : when doing quick zooming or panning (less than 300ms in between two actions), I append all actions in order to make one final undo action.

Hope this helps

Hi Didier,

Sure, the final solution will probably cache the updates and only call it every N ms, but it still does not hurt to investigate the overall performance. Here is some valuable findings: they way TheIDE does it is 4+ times slower than my old solution.

And quite honestly, the performance of LineEdit is downright random. As an example, this is the shortest solution I found:

edtPane.Insert(edtPane.Total(), str); // I added a test getter for total

This executes in 1.4 seconds.

But this:
edtPane.Insert(edtPane.Total(), str);
edtPane.ScrollEnd();

Executes in 300 ms. Without any caching. And has the advantage of scrolling the LineEdit to the end. This solution is 36 times faster than the one inspired from TheIDE/Console.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Didier on Wed, 23 Dec 2015 18:47:49 GMT
View Forum Message <> Reply to Message

Well, it looks like adding edtPane.ScrollEnd() after insert prevents from drawing all text (added and present) : it probably only draws the last visible lines.

Just a guess:
 * You're solution : render time is almost constant (mostly not linked to text size).
 * Original solution (without "ScrollEnd()") : display time directly linked to total text length

---

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Wed, 23 Dec 2015 23:03:45 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Mon, 21 December 2015 12:39Yeah, using the code from Console that manipulates selection and uses Paste I got 11 seconds.

If you provide me with testcase, I can either advise what you are doing wrong, or, if nothing, optimize LineEdit :)

IMO there is not technical reason why that should be slow.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Novo on Thu, 24 Dec 2015 02:11:49 GMT
View Forum Message <> Reply to Message

mirek wrote on Wed, 23 December 2015 18:03cbpporter wrote on Mon, 21 December 2015 12:39Yeah, using the code from Console that manipulates selection and uses Paste I got 11 seconds.

If you provide me with testcase, I can either advise what you are doing wrong, or, if nothing, optimize LineEdit :)

IMO there is not technical reason why that should be slow.

A test case is very simple. Try to compile in TheIDE code, which has a lot of errors/warnings, and TheIDE will frieze for several seconds. I do not know which part of TheIDE is responsible for that (parser, console, or new grid control with error messages), but profiling can help to figure that out. I spotted the problem when tried to compile one of my projects with a wrong compiler.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Thu, 24 Dec 2015 04:49:38 GMT
View Forum Message <> Reply to Message

Novo wrote on Thu, 24 December 2015 03:11mirek wrote on Wed, 23 December 2015 18:03cbpporter wrote on Mon, 21 December 2015 12:39Yeah, using the code from Console that manipulates selection and uses Paste I got 11 seconds.

If you provide me with testcase, I can either advise what you are doing wrong, or, if nothing, optimize LineEdit :)

IMO there is not technical reason why that should be slow.

A test case is very simple. Try to compile in TheIDE code, which has a lot of errors/warnings, and TheIDE will frieze for several seconds. I do not know which part of TheIDE is responsible for that (parser, console, or new grid control with error messages), but profiling can help to figure that out. I spotted the problem when tried to compile one of my projects with a wrong compiler.

I was fixing that about year ago. Is it still an issue with current release? (Just want to be sure before I dive into it again...)

Anyway, the issue back then was not with LineEdit, but problem with pipe.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Novo on Thu, 24 Dec 2015 05:16:42 GMT

View Forum Message <> Reply to Message

mirek wrote on Wed, 23 December 2015 23:49I was fixing that about year ago. Is it still an issue with current release? (Just want to be sure before I dive into it again...)

Anyway, the issue back then was not with LineEdit, but problem with pipe.
This happened to me a couple of weeks ago. I ran a Windows version of TheIDE. Unfortunately, I cannot profile anything by myself because my Linux machine is broken. Well, I can try to run Linux in a VM ...

BTW, could you please fix e-mail notifications for me? I'm not having them since you upgraded forum's software. This is very inconvenient and I'm missing post I could've answered.

Thanks.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Thu, 24 Dec 2015 09:25:09 GMT

View Forum Message <> Reply to Message

mirek wrote on Thu, 24 December 2015 01:03cbpporter wrote on Mon, 21 December 2015 12:39Yeah, using the code from Console that manipulates selection and uses Paste I got 11 seconds.

If you provide me with testcase, I can either advise what you are doing wrong, or, if nothing, optimize LineEdit :)

IMO there is not technical reason why that should be slow.
Sorry, I won't be able to provide test-cases probably until January. But if you are curious, you could run a program that print numbers form 0 to 100000 in a for loop, each followed by a '\n', an time how much it takes for TheIDE to capture and display the output.

There will be differences based on relative computer power, but for me with my solution it takes 300 ms, so that's a starting point. If it is hundreds of ms, it is fine. If it is 11 seconds, like in my case, there might be some problems :).

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by deep on Sat, 26 Dec 2015 11:55:57 GMT

View Forum Message <> Reply to Message

Generally in this scenario I use Append to string then Paste it to LineEdit. (Paste append the text to LineEdit);

 LineEdit le1;

String s;

 loop { s << "Your values and text\n" ;}
 le1.Paste(s.ToWString()),

Even if we append the LineEdit with 100000 line in 100 sec.
This scroll speed also is fast to read and comprehend the values as it scrolls.

When value generation is very fast then I use 2 strings. And pasting to Line edit from another thread.

String s0,s1;

Start with putting values to s0
When s0 is full start filling s1. Paste s0 to LineEdit.
When s1 is full swith filling the s0. Paste s1 to LineEdit.

It is possible to switch strings on time basis or on Number of lines in a string basis. Probably both together.

depends on requirement.

This works for me. I work with real fast data capture.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Novo on Sat, 09 Jan 2016 03:50:48 GMT
View Forum Message <> Reply to Message

I posted profiling info here.

Thanks.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Tue, 12 Jan 2016 11:59:30 GMT
View Forum Message <> Reply to Message

Novo wrote on Sat, 09 January 2016 04:50I posted profiling info here.

Thanks.

I have checked it and if I understand it right, the issue has nothing to do with LineEdit. The

problem there seems to be related to capturing errors for that nice new (since the last year) error window and with some checks invoked by ProcessEvents. I believe both could be fixed quite easily.

Mirek

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by cbpporter on Wed, 13 Jan 2016 14:31:33 GMT
View Forum Message <> Reply to Message

I don't really know what that profiling info is about, but for what I need it to do (capture medium size outputs in LineEdit from a different thread), the method I detailed above:

edtPane.Insert(edtPane.GetTotal(), str);
edtPane.ScrollEnd();

Is the only one that has great performance, so I consider this issue closed for me. I get a 10-20% penalty for capturing data to LineEdit vs. not capturing, so that is perfectly acceptable.

Copy and pasting the code from TheIDE behaved horribly though, maybe I did something wrong.

And there seems to be nothing wrong with LineEdit, as long as you use it right.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by Novo on Thu, 14 Jan 2016 03:56:55 GMT
View Forum Message <> Reply to Message

mirek wrote on Tue, 12 January 2016 06:59Novo wrote on Sat, 09 January 2016 04:50I posted profiling info here.

Thanks.

I have checked it and if I understand it right, the issue has nothing to do with LineEdit. The problem there seems to be related to capturing errors for that nice new (since the last year) error window and with some checks invoked by ProcessEvents. I believe both could be fixed quite easily.

Mirek

Yes. It looks like the problem is not related to LineEdit directly. As I wrote previously " I do not know which part of TheIDE is responsible for that (parser, console, or new grid control with error messages)"
I cannot see anything related to the error window. It is all about Console which spends almost all

time in Ide::FindLineError.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Thu, 14 Jan 2016 06:52:58 GMT
View Forum Message <> Reply to Message

Novo wrote on Thu, 14 January 2016 04:56mirek wrote on Tue, 12 January 2016 06:59Novo wrote on Sat, 09 January 2016 04:50I posted profiling info here.

Thanks.

I have checked it and if I understand it right, the issue has nothing to do with LineEdit. The problem there seems to be related to capturing errors for that nice new (since the last year) error window and with some checks invoked by ProcessEvents. I believe both could be fixed quite easily.

Mirek

Yes. It looks like the problem is not related to LineEdit directly. As I wrote previously " I do not know which part of TheIDE is responsible for that (parser, console, or new grid control with error messages)"
I cannot see anything related to the error window. It is all about Console which spends almost all time in Ide::FindLineError.

That FindLineError is called to determine whether line just added to console is line with error and should be put to error table (not technically a window). ide/Errors.cpp.

Subject: Re: How to do fast Append & scroll in LineEdit with MT?
Posted by mirek on Mon, 18 Jan 2016 09:00:10 GMT
View Forum Message <> Reply to Message

Optimized...