Smuxi - Bug # 211: /names command takes ages on big channels

Status:	Closed	**Priority:**	Normal
Author:	Mirco Bauer	**Category:**	Frontend GNOME
Created:	10/14/2009	**Assigned to:**	Mirco Bauer
Updated:	12/31/2009	**Due date:**	
Subject:	/names command takes ages on big channels		
Description:	The /names command takes around 10 seconds to complete on a channel with 400 members. Something is making it really slow.		

Associated revisions

12/31/2009 04:31 AM - Mirco Bauer

Don't use the TextBuffer.Changed event to refresh the text marks and delete line checks as it's raised way too often for that causing major slowdown for messages with lots of color information. /names command went from 9 seconds down to 430 ms on a channel with 333 users. (closes: #211)

History

11/29/2009 05:00 PM - Mirco Bauer

- Target version set to 0.7

12/30/2009 01:23 PM - Mirco Bauer

The color contrast algo seems to have high CPU usage, this should be profiled and optimized.

12/31/2009 03:42 AM - Mirco Bauer

It's NOT the contrast algo as I thought, so it must be something else:

<pre>

GetBestTextColor(): ##00 ##000000|#EBEBEB=#000000 took: 2.05 ms

GetBestTextColor(): ##01 ##000000|#EBEBEB=#000000 took: 0.09 ms

GetBestTextColor(): ##02 ##FF0000|#EBEBEB=#F50000 took: 0.15 ms

GetBestTextColor(): ##03 ##00FF00|#EBEBEB=#008700 took: 0.18 ms

GetBestTextColor(): ##04 ##0000FF|#EBEBEB=#0000FF took: 0.01 ms

GetBestTextColor(): ##05 ##FF00FF|#EBEBEB=#D700D7 took: 0.05 ms

GetBestTextColor(): ##06 ##FFFF00|#EBEBEB=#7D7D00 took: 0.14 ms

GetBestTextColor(): ##07 ##FFFFFF|#EBEBEB=#737373 took: 0.15 ms

GetBestTextColor(): ##08 ##1E0DD6|#EBEBEB=#1E0DD6 took: 0.01 ms

GetBestTextColor(): ##09 ##1E0DD6|#EBEBEB=#1E0DD6 took: 0.00 ms

GetBestTextColor(): ##10 ##219207|#EBEBEB=#178800 took: 0.02 ms

GetBestTextColor(): ##11 ##429FB0|#EBEBEB=#248192 took: 0.04 ms

GetBestTextColor(): ##12 ##352878|#EBEBEB=#352878 took: 0.02 ms

GetBestTextColor(): ##13 ##52248B|#EBEBEB=#52248B took: 0.01 ms

GetBestTextColor(): ##14 ##603D40|#EBEBEB=#603D40 took: 0.01 ms

GetBestTextColor(): ##15 ##872F56|#EBEBEB=#872F56 took: 0.01 ms

GetBestTextColor(): ##16 ##97608C|#EBEBEB=#97608C took: 0.01 ms

GetBestTextColor(): ##17 ##055A4F|#EBEBEB=#055A4F took: 0.01 ms

GetBestTextColor(): ##18 ##05730C|#EBEBEB=#05730C took: 0.01 ms

GetBestTextColor(): ##19 ##A45DDA|#EBEBEB=#9A53D0 took: 0.02 ms

GetBestTextColor(): ##20 ##279C2A|#EBEBEB=#138816 took: 0.02 ms

GetBestTextColor(): ##21 ##D24F81|#EBEBEB=#C84577 took: 0.02 ms

GetBestTextColor(): ##22 ##45D6FA|#EBEBEB=#007CA0 took: 0.09 ms

GetBestTextColor(): ##23 ##31DD0B|#EBEBEB=#008D00 took: 0.06 ms

GetBestTextColor(): ##24 ##429FB0|#EBEBEB=#248192 took: 0.00 ms

GetBestTextColor(): ##25 ##05FC8F|#EBEBEB=#008E21 took: 1.94 ms

GetBestTextColor(): ##26 ##C1FFEF|#EBEBEB=#3F7D6D took: 0.11 ms

GetBestTextColor(): ##27 ##C1FFEF|#EBEBEB=#3F7D6D took: 0.00 ms

GetBestTextColor(): ##28 ##E4DA22|#EBEBEB=#807600 took: 0.09 ms
GetBestTextColor(): ##29 ##000000|#2E3436=#828282 took: 0.10 ms
GetBestTextColor(): ##30 ##000000|#2E3436=#828282 took: 0.00 ms
GetBestTextColor(): ##31 ##FF0000|#2E3436=#FF3232 took: 0.04 ms
GetBestTextColor(): ##32 ##00FF00|#2E3436=#00FF00 took: 0.01 ms
GetBestTextColor(): ##33 ##0000FF|#2E3436=#7878FF took: 0.09 ms
GetBestTextColor(): ##34 ##FF00FF|#2E3436=#FF00FF took: 0.01 ms
GetBestTextColor(): ##35 ##FFFF00|#2E3436=#FFFF00 took: 0.01 ms
GetBestTextColor(): ##36 ##FFFFFF|#2E3436=#FFFFFF took: 0.01 ms
GetBestTextColor(): ##37 ##1E0DD6|#2E3436=#8271FF took: 0.08 ms
GetBestTextColor(): ##38 ##1E0DD6|#2E3436=#8271FF took: 0.00 ms
GetBestTextColor(): ##39 ##219207|#2E3436=#2B9C11 took: 0.01 ms
GetBestTextColor(): ##40 ##429FB0|#2E3436=#429FB0 took: 0.01 ms
GetBestTextColor(): ##41 ##352878|#2E3436=#8578C8 took: 0.05 ms
GetBestTextColor(): ##42 ##52248B|#2E3436=#A274DB took: 0.06 ms
GetBestTextColor(): ##43 ##603D40|#2E3436=#A68386 took: 0.05 ms
GetBestTextColor(): ##44 ##872F56|#2E3436=#C36B92 took: 0.06 ms
GetBestTextColor(): ##45 ##97608C|#2E3436=#AB74A0 took: 0.02 ms
GetBestTextColor(): ##46 ##055A4F|#2E3436=#41968B took: 0.05 ms
GetBestTextColor(): ##47 ##05730C|#2E3436=#2D9B34 took: 0.04 ms
GetBestTextColor(): ##48 ##A45DDA|#2E3436=#AE67E4 took: 0.01 ms
GetBestTextColor(): ##49 ##279C2A|#2E3436=#279C2A took: 0.01 ms
GetBestTextColor(): ##50 ##D24F81|#2E3436=#DC598B took: 0.02 ms
GetBestTextColor(): ##51 ##45D6FA|#2E3436=#45D6FA took: 0.01 ms
GetBestTextColor(): ##52 ##31DD0B|#2E3436=#31DD0B took: 0.01 ms
GetBestTextColor(): ##53 ##429FB0|#2E3436=#429FB0 took: 0.00 ms
GetBestTextColor(): ##54 ##05FC8F|#2E3436=#05FC8F took: 0.01 ms
GetBestTextColor(): ##55 ##C1FFEF|#2E3436=#C1FFEF took: 0.00 ms
GetBestTextColor(): ##56 ##C1FFEF|#2E3436=#C1FFEF took: 0.00 ms
GetBestTextColor(): ##57 ##E4DA22|#2E3436=#E4DA22 took: 0.01 ms
</pre>

12/31/2009 03:57 AM - Mirco Bauer

It's not the remoting transport either, the call of AddMessage() itself on the widget really takes almost 10 seconds on a channel with 333 users (tested with #debian @ OFTC):

<pre>
2009-12-31 04:55:55,012 [Main] DEBUG Smuxi.Frontend.Gnome.GnomeUI - _AddMessageToChat(): chatView.AddMessage() took: 9799.19 ms
</pre>

12/31/2009 04:02 AM - Mirco Bauer

2nd /names command took even longer, wtf?

<pre>
2009-12-31 05:01:30,889 [Main] DEBUG Smuxi.Frontend.Gnome.GnomeUI - _AddMessageToChat(): chatView.AddMessage() took: 21413.43 ms
</pre>

12/31/2009 04:04 AM - Mirco Bauer

3rd /names call a bit better but still slower than 1st:

<pre>
2009-12-31 05:03:09,607 [Main] DEBUG Smuxi.Frontend.Gnome.GnomeUI - _AddMessageToChat(): chatView.AddMessage() took: 14627.58 ms
</pre>

I believe it's either the TextView itself or how I add the formatted / colored pieces to it... or something else that gets triggered (events?) slow it down.

12/31/2009 04:12 AM - Mirco Bauer

Looks like its the ChatView._OnTextBufferChanged() that slows it really down. Every buffer insert triggers that method, and /names does lots of buffer inserts for all the color tags. /names on #smuxi went down from 35ms to 5ms :)

12/31/2009 04:27 AM - Mirco Bauer

Yep, that was the issue. _OnTextBufferChanged() was refreshing text marks and also checking if lines need to be removed and that for every single buffer change (adding one line might do hundreds of changes). The code was simply moved from _OnTextBufferChange to OnMessageTextViewMessageAdded, making it executed once per message was added.

12/31/2009 04:28 AM - Mirco Bauer

Oh and that 9 seconds went down to 430 ms on the first call and 130 ms in the second call :-)

12/31/2009 04:35 AM - Mirco Bauer

- Status changed from New to Closed

- % Done changed from 0 to 100

Applied in changeset commit:"84207473f0f578cab268e97eff8d3da57bed9ec0".