

QGIS Application - Bug report #4514

New symbology fails to render "problematic" lines at high zoom levels

2011-11-12 01:44 AM - Sandro Santilli

Status:	Closed
Priority:	Normal
Assignee:	
Category:	Symbology
Affected QGIS version:	master
Operating System:	
Pull Request or Patch Supplied:	
Crashes QGIS or corrupts data:	
Regression?:	No
Easy fix?:	No
Resolution:	fixed
Copied to github as #:	14433

Description

The following WKT has a few rendering problems.

It is a V shaped figure. One element is the full V, the other two are the two wings.

The bottom node is coincident for the two wings, but it's moved on the left for the full V.

Beside other rendering problems, for which I'll file another bug, this one is about a different treatment between old symbology and new symbology. To reproduce:

Applying old symbology and zooming on the right wing until you see two lines.

Then switch to new symbology and apply it. The lines will disappear.

This is with a recent build from master, and specifically commit be9bdeb

```
MULTILINESTRING(  
(  
832709.937 816560.25,  
832705.813 816470.25,  
832661.937 816561.875  
),  
(  
832705.812 816470.25,  
832709.937 816560.25  
(  
832661.938 816561.875,  
832705.813 816470.25  
))
```

History

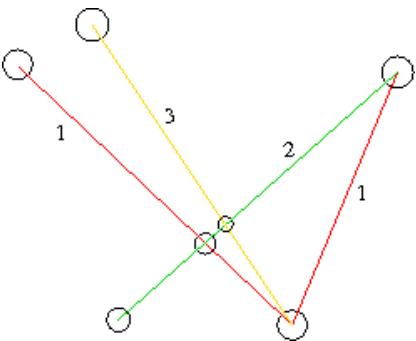
#1 - 2011-11-12 02:13 AM - Sandro Santilli

- File topotest.png added

I was wrong about the actual shape.

Now made a drawing to represent it.

#2 - 2011-11-12 02:19 AM - Sandro Santilli



#3 - 2011-12-09 08:42 AM - Paolo Cavallini

- Category set to *Symbology*

#4 - 2011-12-16 02:10 PM - Giovanni Manghi

- Target version set to *Version 1.7.4*

#5 - 2012-03-06 01:13 AM - Sandro Santilli

- Affected QGIS version set to *master*

- Status changed from *Open* to *Closed*

- Resolution set to *invalid*

- Crashes QGIS or corrupts data set to *No*

Well, since I was wrong about the shape this ticket is invalid.

#6 - 2012-03-06 01:18 AM - Sandro Santilli

- Status changed from *Closed* to *Reopened*

- Resolution deleted (*invalid*)

Actually, since old and new symbology behave differently this is likely still a valid bug.

I was looking for this ticket because I had another similar occurrence of it.

Lines disappearing on zoom in.

I'm not sure, but it could have to do with lines clipping done in 17e864bb99c5c6b2e1f0493a952870f183d9d9c0

which I think addressed #4088 in addition to the reported #4083.

#7 - 2012-03-06 01:43 AM - Sandro Santilli

Here's another input showing the problem:

```
LINESTRING(556267.657697 144886.732981, 556267.657697 144886.732981, 556267.517494 144888.366260)
LINESTRING(556235.246910 144885.298021, 556235.246910 144885.298021, 556254.670000 144886.620000, 556267.657697
144886.732981)
LINESTRING(556267.657697 144886.732981, 556310.040000 144888.170000, 556317.280000 144888.990000, 556328.490000
144888.800000, 556329.964330 144889.104630, 556336.040000 144890.360000, 556354.074605 144896.287433, 556362.830000
144899.280000)
```

I hit another bug while trying to reproduce (#5120) because the problem can't be reproduced with old symbology (or maybe w/out labels, not sure yet).

#8 - 2012-03-06 05:17 AM - Sandro Santilli

Ok, with #5120 fixed the input in previous commit is enough to reproduce *this* bug. Doesn't need labels, just switching to the new symbology. Zoom on the intersection of the 3 lines to see them moving.

#9 - 2012-03-06 05:36 AM - Sandro Santilli

Note: it renders fine up to scale 1:10, fails at 1:5 and higher

#10 - 2012-04-16 06:32 AM - Paolo Cavallini

- *Target version changed from Version 1.7.4 to Version 1.8.0*

#11 - 2012-06-17 12:35 PM - Serge Dikiy

Hi, @strk

<https://github.com/qgis/Quantum-GIS/blob/master/src/core/symbology-ng/qgsrendererv2.cpp#L79>

New symbology use [Line clipping](#) only if (nPoints > 100)

I use a very simple patch to fix the problem :-)

<https://github.com/qgis/Quantum-GIS/pull/179/files>

#12 - 2012-06-17 01:02 PM - Marco Hugentobler

- *Status changed from Reopened to Closed*

The fix is applied, thank you Serge.

The symbology-ng line clipping has been made to overcome performance issues with large lines (the Qt clipping itself is too slow). If you say that it also fixes precision problems with shorter lines, that's fine.

#13 - 2012-06-18 12:36 AM - Sandro Santilli

- *Status changed from Closed to Reopened*

I'd like to have a chance to actually test this.

My impression was the "clipping" was the cause of the bug, not the cure.

That's based on the fact that "old" symbology (surely not clipping) doesn't suffer from the bug.

#14 - 2012-06-18 12:44 AM - Marco Hugentobler

Thanks for your availability to do more tests, that will be helpful.

Btw. the old symbology clips linestrings too, but with a different algorithm (see function QgsVectorLayer::drawLineString)

#15 - 2012-06-18 01:11 AM - Serge Dikiy

I think this is a bug Qt.

This bug appears when the ratio of the line segment to its thickness exceeds a certain value.

Bug is not reproduced for the line width greater than 0,3. (I use dataset from original post)

Bug is not reproduced for the dotted line style.

"clipping" reduces the length of the line before rendering.

#16 - 2012-06-18 06:22 AM - Sandro Santilli

- Resolution set to fixed

- Status changed from Reopened to Closed

- % Done changed from 0 to 50

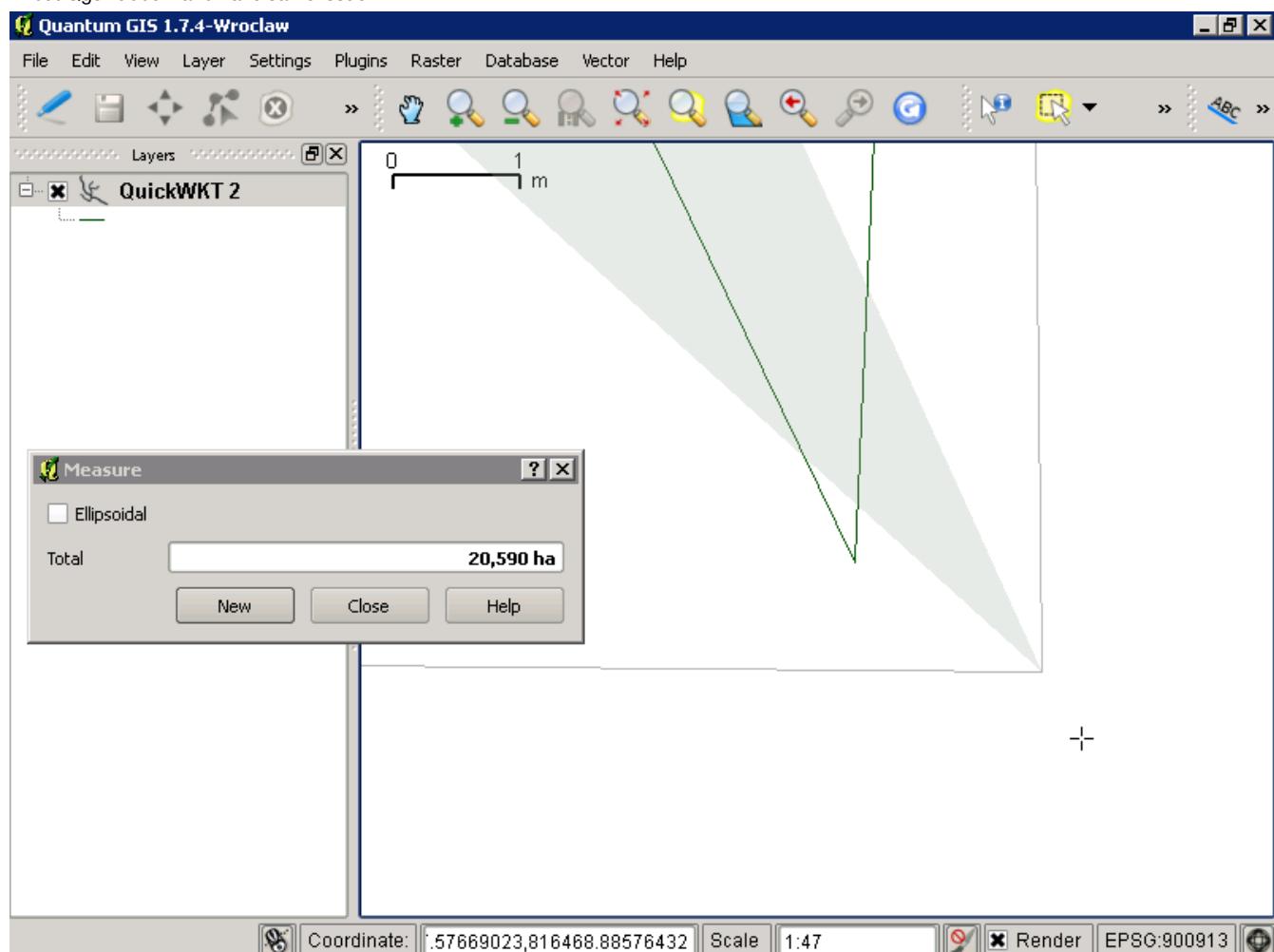
Ok, tested. 5b09b96 looks fine to me. Thanks a lot.

Would be great to have this case under regression testing.

#17 - 2012-06-18 06:41 AM - Serge Dikiy

- File QgsRubberBand.PNG added

...but QgsRubberBand have same issue.



#18 - 2012-06-18 07:04 AM - Sandro Santilli

do you mean there are still differences between old and new symbology ? That's my main concern, hoping to see one fully replacing the other asap (so to all focus on a single implementation).

#19 - 2012-06-26 05:28 AM - Sandro Santilli

For the record: it was sorted out in IRC that the RubberBand thing has nothing to do with this ticket and that it will eventually need a separate ticket to handle.

Files

topotest.png	3.45 KB	2011-11-12	Sandro Santilli
QgsRubberBand.PNG	35.5 KB	2012-06-18	Serge Dikiy