QGIS Application - Bug report #19638
CSV: numeric header gets a trailing underscore + 1

2018-08-17 08:15 AM - Tobias Wendorff

Status:	Closed			
Priority:	Normal			
Assignee:				
Category:	Data Provider/Delimited Text			
Affected QGIS version:	3.2.1		**Regression?:**	No
Operating System:	Microsoft Windows 7, 64-bit		**Easy fix?:**	No
Pull Request or Patch supplied:	Yes		**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	No		**Copied to github as #:**	27465

Description

When loading a CSV / delimited text with a header like this "id;1;2;3;4", the numeric header gets a trailing underscore and a "1" =>
"id;1_1;2_1;3_1;4_1". So the fieldnames are broken.
Expected behavior: "id;1;2;3;4"

All versions >= 3.1 are affected.

Associated revisions

Revision 379652d2 - 2018-08-27 03:55 AM - Andrea Giudiceandrea

delimited text: let field names be numerical value

fixes #19638 "CSV: numeric header gets a trailing underscore + 1"
fixes #13187 "Numeric FIELD names are lost on CSV import"

Revision a5fd8137 - 2018-08-31 02:32 AM - Andrea Giudiceandrea

delimited text: let field names be numerical value

fixes #19638 "CSV: numeric header gets a trailing underscore + 1"
fixes #13187 "Numeric FIELD names are lost on CSV import"

(cherry picked from commit 379652d2022c5373d679d0d0edddfa53f27b9453)

History

#1 - 2018-08-19 11:16 AM - Giovanni Manghi

- Status changed from Open to Feedback

- Crashes QGIS or corrupts data changed from Yes to No

- Regression? changed from Yes to No

On 2.18/LTR the same happens?

#2 - 2018-08-19 11:27 AM - Tobias Wendorff

Giovanni Manghi wrote:

> *On 2.18/LTR the same happens?*

I don't know, I'm not using 2.x any longer. But In my eyes, a change from "1" to "1_1" is data corruption. Data shouldn't be modified on import, especially not data coming from plaintext.

#3 - 2018-08-19 11:29 AM - Giovanni Manghi

- Status changed from Feedback to Open

Tobias Wendorff wrote:

> *Giovanni Manghi wrote:*
>
> > *On 2.18/LTR the same happens?*
>
> *I don't know, I'm not using 2.x any longer. But In my eyes, a change from "1" to "1_1" is data corruption. Data shouldn't be modified on import, especially not data coming from plaintext.*

is not data corruption because your original dataset is not corrupted. If you want this to be tagged as "regression" you should check if on 2.18 works as expected.

#4 - 2018-08-19 11:34 AM - Tobias Wendorff

Giovanni Manghi wrote:

> *is not data corruption because your original dataset is not corrupted.*

The header is part of the dataset. If you export it back to CSV, you'll get "1_1" instead of "1". Please don't modify ANY data if not needed.

> *If you want this to be tagged as "regression" you should check if on 2.18 works as expected.*

I thought "regression" would only apply to current main version: 3.0, 3.1, 3.2 etc. Sorry for that.

#5 - 2018-08-20 11:04 AM - Giovanni Manghi

Tobias Wendorff wrote:

> *Giovanni Manghi wrote:*
>
> > *is not data corruption because your original dataset is not corrupted.*
>
> *The header is part of the dataset. If you export it back to CSV, you'll get "1_1" instead of "1". Please don't modify ANY data if not needed.*

still the original dataset is not modified/lost. For data corruption we usually mean unrecoverable data loss. Here the issue in different: a wrong result from the manipulation of a dataset (that is unchanged).

I thought "regression" would only apply to current main version: 3.0, 3.1, 3.2 etc. Sorry for that.

Regression is anything that stopped to work and that used to work in any previous release. LTR releases are usually used as reference for comparison (at lease is what I do).

#6 - 2018-08-20 04:02 PM - Andrea Giudiceandrea

I've tested the issue (a csv with the header "id;1;2;3;4") with the following QGIS version:

- 2.18.19 behaves as reported by Tobias Wendorff for 3.2.1: imported layer field names "id;1_1;2_1;3_1;4_1"
- 1.9.0 alpha (bb0b978) and 1.8.0 behave as Tobias think it should be: imported layer field names "id;1;2;3;4"

If the csv header is e.g. "id;2;3;4;5" the issue does not occur.

I suppose this issue/feature was introduced with commit:5e4f4f73bad29b7d88dc5fb2e64d264d325f7a27 by Chris Crook and is present since QGIS 2.0

https://github.com/qgis/QGIS/blob/master/src/providers/delimitedtext/qgsdelimitedtextfile.cpp#L37-L38
```
    // field_ is optional in following regexp to simplify QgsDelimitedTextFile::fieldNumber()
      , mDefaultFieldRegexp( "^(?:field_)?(\\d+)$", Qt::CaseInsensitive )
```

https://github.com/qgis/QGIS/blob/master/src/providers/delimitedtext/qgsdelimitedtextfile.cpp#L418-L424
```
    // If the name looks like a default field name (field_##), then it is
      // valid if the number matches its column number..
      else if ( mDefaultFieldRegexp.indexIn( name ) == 0 )
    {
        int col = mDefaultFieldRegexp.capturedTexts().at( 1 ).toInt();
        nameOk = col  fieldNo;
      }
```

It seems the issue is caused by the fact that "field_" is optional in mDefaultFieldRegexp, so "mDefaultFieldRegexp.indexIn(name) 0" is true if column name began with "field_" OR **if it contains numerical digits only**. In the latter case, **if the number doesn't match its column number, then "_1" is appended as a suffix to the column name**.

We can fix the issue letting ""field_" not optional in mDefaultFieldRegexp:
```
    mDefaultFieldRegexp( "^(?:field_)(\\d+)$", Qt::CaseInsensitive )
```

but I don't know if there was a valid reason for not doing it then (in the comment it's written that the reason is "to simplify QgsDelimitedTextFile::fieldNumber()", although I can not find this "fieldNumber()")

See also #13187 that seems not really fixed.

#7 - 2018-08-20 04:47 PM - Tobias Wendorff

Giovanni Manghi wrote:

> *Regression is anything that stopped to work and that used to work in any previous release. LTR releases are usually used as reference for comparison (at lease is what I do).*

Thanks! Please add this to a FAQ. I'll install LTR in parallel for the next reports (I've still some on my list).

#8 - 2018-08-20 06:31 PM - Giovanni Manghi

Andrea Giudiceandrea wrote:

> I've tested the issue (a csv with the header "id;1;2;3;4") with the following QGIS version:
>
> - 2.18.19 behaves as reported by Tobias Wendorff for 3.2.1: imported layer field names "id;1_1;2_1;3_1;4_1"
> - 1.9.0 alpha (bb0b978) and 1.8.0 behave as Tobias think it should be: imported layer field names "id;1;2;3;4"
>
> If the csv header is e.g. "id;2;3;4;5" the issue does not occur.
>
> I suppose this issue/feature was introduced with commit:5e4f4f73bad29b7d88dc5fb2e64d264d325f7a27 by Chris Crook and is present since QGIS 2.0
>
> https://github.com/qgis/QGIS/blob/master/src/providers/delimitedtext/qgsdelimitedtextfile.cpp#L37-L38
>
> // field_ is optional in following regexp to simplify QgsDelimitedTextFile::fieldNumber()
> , mDefaultFieldRegexp("^(?:field_)?(\\d+)$", Qt::CaseInsensitive)
>
> https://github.com/qgis/QGIS/blob/master/src/providers/delimitedtext/qgsdelimitedtextfile.cpp#L418-L424
>
> // If the name looks like a default field name (field_##), then it is
> // valid if the number matches its column number..
> else if (mDefaultFieldRegexp.indexIn(name) == 0){
> int col = mDefaultFieldRegexp.capturedTexts().at(1).toInt();
> nameOk = col fieldNo;
> }
>
> It seems the issue is caused by the fact that "field_" is optional in mDefaultFieldRegexp, so "mDefaultFieldRegexp.indexIn(name) 0" is true if column name began with "field_" OR **if it contains numerical digits only**. In the latter case, **if the number doesn't match its column number, then "_1" is appended as a suffix to the column name**.
>
> We can fix the issue letting ""field_" not optional in mDefaultFieldRegexp:
>
> mDefaultFieldRegexp("^(?:field_)(\\d+)$", Qt::CaseInsensitive)
>
> but I don't know if there was a valid reason for not doing it then (in the comment it's written that the reason is "to simplify QgsDelimitedTextFile::fieldNumber()", although I can not find this "fieldNumber()")

Hi, thanks for this comments. You should really comment among the lines on Github, this way there is a much bigger chance a developer will notice the comment and read your suggestions. Even better if you could come up with a patch (also submitted on github). Thanks!

#9 - 2018-08-21 12:40 PM - Andrea Giudiceandrea

PR submitted https://github.com/qgis/QGIS/pull/7671

#10 - 2018-08-21 07:27 PM - Giovanni Manghi

- Pull Request or Patch supplied changed from No to Yes

Andrea Giudiceandrea wrote:

> PR submitted *https://github.com/qgis/QGIS/pull/7671*

nice, thanks!

#11 - 2018-08-27 03:55 AM - Andrea Giudiceandrea

- Status changed from Open to Closed

- % Done changed from 0 to 100

Applied in changeset commit:qgis|379652d2022c5373d679d0d0edddfa53f27b9453.

#12 - 2018-08-27 07:27 AM - Tobias Wendorff

Andrea Giudiceandrea wrote:

> *Applied in changeset commit:qgis|379652d2022c5373d679d0d0edddfa53f27b9453.*

Thanks to all.

#13 - 2018-08-29 11:38 AM - Giovanni Manghi

- Resolution set to fixed/implemented