QGIS Application - Bug report #14262
Overflow on primary key with negative values; cannot save edits

2016-02-09 08:24 AM - Sandro Santilli

Status:	Closed		
Priority:	Normal		
Assignee:	Sandro Santilli		
Category:	Data Provider/PostGIS		
Affected QGIS version:	2.14.3	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	Yes	**Resolution:**	fixed/implemented
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	22258

Description

Spin off of #13958

Create a simple PostGIS table:

```
CREATE table my_lines(gid integer primary key, col integer);
SELECT AddGeometryColumn('my_lines', 'geom', 4326, 'MULTILINESTRING', 2);

INSERT INTO my_lines(gid, col, geom)
SELECT -1, 2, 'SRID=4326;MULTILINESTRING((0 0, 1 1))';
```

Note that the primary key has a negative value -1, which should be completely fine and previous versions of QGIS were OK with this.

This table can be added to QGIS, which shows the simple geometry on the main canvas normally, and the Layer properties correctly describes both gid and col fields as int4.

However, the attribute table shows the values of gid and col as ERROR, and clicking the feature with the Identify Features tool show gid of 4294967295, which is the overflow version of an unsigned integer value -1. Perhaps the primary key was internally cast in QGIS to unsigned long?

Editing the layer, e.g. with the Node tool fail on save, with this message:

```
src/providers/postgres/qgspostgresprovider.cpp: 2322: (changeGeometryValues) [1279ms] entering.
src/providers/postgres/qgspostgresprovider.cpp: 2395: (changeGeometryValues) [0ms] updating: UPDATE "public"."my_lines"
SET "geom"=st_multi(st_geom
fromwkb($1::bytea,4326)) WHERE "gid"=$2
src/providers/postgres/qgspostgresprovider.cpp: 2405: (changeGeometryValues) [1ms] iterating over the map of changed
geometries...
src/providers/postgres/qgspostgresprovider.cpp: 2411: (changeGeometryValues) [0ms] iterating over feature id 4294967295
src/core/qgsvectordataprovider.cpp: 560: (pushError) [0ms] PostGIS error while changing geometry values: ERROR:  value
"4294967295" is out of range
 for type integer

src/providers/postgres/qgspostgresprovider.cpp: 2501: (changeGeometryValues) [0ms] leaving.
src/core/qgsmessagelog.cpp: 45: (logMessage) [1ms] 2016-02-09T17:20:03 [1] Commit errors:
  ERROR: 1 geometries not changed.
```

Provider errors:
PostGIS error while changing geometry values: ERROR: value "4294967295" is out of range for type integer

Affected: master branch 2.14.0dev (commit:b9726d7285733c27d42456c115e28d5a37f3e0be)

Note that 2.8.3 crashes on first edit attempt. The crash is fixed in current master.

History

#1 - 2016-02-09 01:43 PM - Giovanni Manghi

- Status changed from Open to Feedback

if it worked ok on previous qgis releases this should be tagged as blocker and have 2.14 has target. Cheers!

#2 - 2016-02-09 01:50 PM - Sandro Santilli

As mentioned, it crashed with 2.8.4. I hadn't tested any previous release, but 2.8 being an LTR I wouldn't go any earlier

#3 - 2016-02-09 02:08 PM - Giovanni Manghi

Sandro Santilli wrote:

> *As mentioned, it crashed with 2.8.4. I hadn't tested any previous release, but 2.8 being an LTR I wouldn't go any earlier*

I'm not speaking about the crash, but about the subject of this ticket ("Note that the primary key has a negative value -1, which should be completely fine and previous versions of QGIS were OK with this.").

#4 - 2016-02-09 02:11 PM - Sandro Santilli

Mike Toews did not specify which previous version worked. I did test with 2.8.4 and the feature id is still reported as an unsigned and attribute values reported as ERROR. Do you confirm, Giovanni ?

#5 - 2016-02-10 12:43 AM - Sandro Santilli

I just tried 2.6.1-Brighton and it also displays ERROR in attribute values, max unsigned integer on identify and crashes on edit.

#6 - 2016-02-10 12:50 AM - Sandro Santilli

2.4.0-Chugiak same behavior as 2.6.1-Brighton and 2.8.4

#7 - 2016-02-10 01:10 AM - Sandro Santilli

It looks like QgsPostgresConn::getBinaryInt is responsible of converting the signed integer to an unsigned one:

```
qint64 QgsPostgresConn::getBinaryInt( QgsPostgresResult &queryResult, int row, int col )
{
  quint64 oid;
```

Jurgen, what's the rationale for that ?

#8 - 2016-02-10 02:18 AM - Sandro Santilli

- Pull Request or Patch supplied changed from No to Yes
- % Done changed from 0 to 80
- Target version changed from Future Release - High Priority to Version 2.14
- Assignee set to Sandro Santilli
- Status changed from Feedback to In Progress

Patch ready for review: https://github.com/qgis/QGIS/pull/2777

#9 - 2016-02-16 08:43 AM - Sandro Santilli

Saving edits is handled by not treating integer fields in any special way by
https://github.com/qgis/QGIS/pull/2797

#10 - 2016-02-16 08:48 AM - Sandro Santilli

- Category set to Data Provider/PostGIS

#11 - 2016-02-16 08:54 AM - Sandro Santilli

Saves can be edited with https://github.com/qgis/QGIS/pull/2797, that is using a field map based feature id for the case of integer primary key.

#12 - 2016-06-07 05:05 AM - Sandro Santilli

Current focus for this bug is here: https://github.com/qgis/QGIS/pull/3036

#13 - 2016-06-09 03:03 AM - Sandro Santilli

- % Done changed from 80 to 100
- Target version changed from Version 2.14 to Version 2.16
- Resolution set to fixed/implemented
- Affected QGIS version changed from master to 2.14.3
- Status changed from In Progress to Closed

Fixed as of commit:d1cac84 -- I suspect this didn't work in 2.8 either.

#14 - 2016-06-09 03:23 AM - Sandro Santilli

Question: should this be backported to 2.14 ?

#15 - 2016-06-09 03:26 AM - Sandro Santilli

- Status changed from Closed to Reopened

Will test against 2.14, but given 2.14 is the LTR, it'd be best fixed there (please correct me if I'm wrong)

#16 - 2016-06-09 03:45 AM - Sandro Santilli

- Status changed from Reopened to Closed

Tested: the crash is fixed in 2.14(.3) while the "cannot save edits" part is only fixed in 2.16.
As we aim for stability, I'd not backport the ability to edit (or save edits) in 2.14. Feel free to reopen if you think it should be done instead.