

QGIS Application - Bug report #19564

PostgreSQL sequences not always used when adding feature

2018-08-08 11:24 AM - Nicolas Boisteault

Status:	Open	
Priority:	Normal	
Assignee:		
Category:	Data Provider/PostGIS	
Affected QGIS version:	3.3(master)	Regression?: No
Operating System:	Windows	Easy fix?: No
Pull Request or Patch supplied:	No	Resolution:
Crashes QGIS or corrupts data:	No	Copied to github as #: 27391
Description I have several PostgreSQL layers with no geometries. All have an 'id' column with serial type so they have a primary key and a sequence. When I add a feature, with some of these tables I have 'nextval('schema.id_seq'::regclass)' in the id field (which is the correct behavior) and with some others I have a NULL value. I can't see no difference between the tables with the good behavior and the others. I've also noticed that when I open the 'Attributes form' tab for wrong behavior tables the id field has 'Range' Widget Type proposed. For good behavior tables it is 'Text Edit' Widget Type. Issue also occurs in QGIS 2.18.x and I have tested with 3 different Windows machines. Database is PostgreSQL 9.6 on Ubuntu 16.04. I can give limited access to my database if needed.		

History

#1 - 2018-08-22 07:40 PM - Tudor Bărbăscu

It would be easier to just paste here the SQL code that creates the tables/sequences so your issue can be replicated without access. It's easier for everyone + you may find your solution meanwhile.

#2 - 2018-08-22 09:58 PM - Giovanni Manghi

- Status changed from Open to Feedback

#3 - 2018-09-26 04:08 PM - Nicolas Boisteault

- File *annuaire.sql* added

Same issue with last nightly. I've attached the SQL code. For example table 'choix_secteur_activite' has the good behavior and 'choix_public_cible' doesn't.

#4 - 2018-09-27 09:20 AM - Giovanni Manghi

Nicolas Boisteault wrote:

Same issue with last nightly. I've attached the SQL code. For example table 'choix_secteur_activite' has the good behavior and 'choix_public_cible' doesn't.

they both work the same here (with 'nextval('schema.id_seq'::regclass)'). QGIS Master, pgsq1 10

#5 - 2018-10-08 10:40 AM - Nicolas Boisteault

I have tested on a fresh install of PostgreSQL 9.6 and can't reproduce the issue. As I am the only one having this issue I guess it is not a QGIs one. Thank you for your tests. You can close issue.

#6 - 2018-10-08 11:36 AM - Giovanni Manghi

- Resolution set to invalid
- Status changed from Feedback to Closed

#7 - 2019-03-22 09:22 PM - David Mancini

- File *not_working_gid_sequence.png* added
- File *working_gid_sequence.png* added

I am experiencing the exact same behaviour as described by the original poster. Today for the first time I ran into an issue where the PostgreSQL sequence was not providing the `nextval('name_of_sequence'::regclass)` in the edit form. When comparing against other tables in the database, others perform normally.

Digging a little deeper through the layer properties I also saw that the working table showed my gid SERIAL PRIMARY KEY column as 'Text Edit' widget Type, whereas the non-working table showed it as 'Range' as described by the original poster.

At the database level both these tables were created with the exact same process. Here is the SQL for comparison.

```
CREATE TABLE "temp_BC_ParcelMap_Knockout_20170727_a"
(
    gid serial NOT NULL,
    grmn_type character varying,
    the_geom geometry(Geometry,4326),
    CONSTRAINT "temp_BC_ParcelMap_Knockout_20170727_a_pkey" PRIMARY KEY (gid)
)
WITH (
    OIDS=TRUE
);
ALTER TABLE "temp_BC_ParcelMap_Knockout_20170727_a"
    OWNER TO postgres;
```

```
CREATE TABLE "temp_BC_ParcelMap_Knockout_20190319_a"
(
    gid serial NOT NULL,
    grmn_type character varying,
    the_geom geometry(Geometry,4326),
    CONSTRAINT "temp_BC_ParcelMap_Knockout_20190319_a_pkey" PRIMARY KEY (gid)
)
WITH (
    OIDS=TRUE
);
ALTER TABLE "temp_BC_ParcelMap_Knockout_20190319_a"
    OWNER TO postgres;
```

It seems all the new tables I create are affected by this behaviour, but older tables which have been within the database for the past few years work as expected. Was there any solution to this issue?

I've tested these tables in QGIS 3.6.0 and QGIS 2.18.21 and the bug is affecting both. Our database server is running PostgreSQL 9.4.19 on Debian

8.11.

#8 - 2019-03-28 09:33 AM - Nicolas Boisteault

- Status changed from Closed to Reopened

Thanks David for your feedback. We used trigger to increment sequence as a workaround.

#9 - 2019-03-28 10:10 AM - Alessandro Pasotti

Does it change anything if you change the widget type from TEXT to RANGE or vice-versa?

#10 - 2019-03-28 01:40 PM - Giovanni Manghi

- Status changed from Reopened to Feedback

- Resolution deleted (invalid)

#11 - 2019-03-28 05:34 PM - David Mancini

Unfortunately no, changing the widget type from Range to Text does not resolve the issue.

The only work around I was able to come up with within QGIS 3.6.0 was setting a Default expression as follows: `maximum("gid") + 1`

Not ideal, but does at least allow me to save new features and copy / paste new features in from tables without gid. Having a look into the database I can see that this workaround still does not utilize the currently defined sequence as the Next Value does not continue to grow as new features are added in this way.

#12 - 2019-03-28 05:35 PM - Giovanni Manghi

- Status changed from Feedback to Open

#13 - 2019-05-22 06:40 PM - David Mancini

Nicolas Boisteault wrote:

Thanks David for your feedback. We used trigger to increment sequence as a workaround.

Hello Nicolas. Any chance you can provide the TRIGGER statement you used for your workaround?

Files

annuaire.sql	10.9 KB	2018-09-26	Nicolas Boisteault
not_working_gid_sequence.png	53 KB	2019-03-22	David Mancini
working_gid_sequence.png	46.5 KB	2019-03-22	David Mancini