

QGIS Application - Bug report #20073

Form validation doesn't take care of 2 columns primary key

2018-10-10 08:24 PM - Etienne Trimaille

Status: Closed	
Priority: High	
Assignee:	
Category: Forms	
Affected QGIS version: 3.3(master)	Regression?: Yes
Operating System:	Easy fix?: No
Pull Request or Patch supplied:	Resolution:
Crashes QGIS or corrupts data:	Copied to github as #: 27895
Description	
<p>I have a PostGIS table like this referencing my schemas and my tables:</p> <pre>CREATE TABLE public.sources (schemaname character varying(50) COLLATE pg_catalog."default" NOT NULL, tablename character varying(50) COLLATE pg_catalog."default" NOT NULL, source text COLLATE pg_catalog."default" NOT NULL, id_karugeo character varying(50) COLLATE pg_catalog."default", nom character varying(255) COLLATE pg_catalog."default", CONSTRAINT sources_pkey PRIMARY KEY (schemaname, tablename))</pre> <p>As you can see, I have a primary key composed of 2 columns : schema and table.</p> <p>But in the QGIS form, I can't add a new feature if ONE feature with the given SCHEMA is already in my "source" table. It's like if QGIS is checking my primary key only with the "schema" column not taking care of the composition schema-table.</p> <p>In my screenshot, I can't add my new feature in the table, because I have already some features containing "agro" as schema. If I remove all my features with "agro" in the schema column, then I will be able to add my new feature.</p> <p>QGIS 3.2 and 3.3 are affected.</p>	

Associated revisions

Revision 2826ba34 - 2019-04-07 04:55 AM - Jürgen Fischer

postgres provider: drop implicit not null and unique constraint for columns in compound keys (fixes #20073)

Revision 30e808e3 - 2019-04-17 08:30 AM - Jürgen Fischer

postgres provider: drop implicit not null and unique constraint for columns in compound keys (fixes #20073)

(cherry picked from commit f8e32f7b9e21c2bdee6c80e796f8d448fef19e12)

postgres provider: drop implicit not null and unique constraint for columns in compound keys (fixes #20073)

(cherry picked from commit 30e808e3fc & 267599ba)

History

#1 - 2018-10-10 08:37 PM - Etienne Trimaille

- *File wrong_default_checkbox.gif added*

Moreover, checkbox doesn't keep the same status when we switch from one layer to another one.

Look at my GIF, at the beginning, 2 checkbox are disabled. If I select another layer and come back, the same checkbox are now enabled.

I put this comment in the same ticket, it might be related to the same bug.

#2 - 2018-10-15 12:30 AM - Nyall Dawson

- *Status changed from Open to Feedback*

Etienne -- I'm thinking of removing the force-disabling of these checkboxes so that users can ALWAYS turn off the constraints on QGIS side (if we get them wrong in cases like this). Would that fix the issue for you?

#3 - 2018-10-18 01:59 PM - Mathieu Bossaert

Hi Etienne and Nyall,

same problem here with a 3 columns primary keys (QGIS 3.2 on ubuntu 18.04).

And I think I understood the problem.

QGIS apply a unique constraint to each of the three columns, as it does for a simple (one column) primary key.

I found a workaround to this problem, but not a real clean solution : I defined a rank() integer column (needed by QGIS) as the primary key of the table, and set a UNIQUE constraint over the three columns.

Now I can edit an new line to my table,

#4 - 2018-10-18 10:52 PM - Etienne Trimaille

Yes, I can confirm what Mathieu said. The primary key is done for each column, individually, not checking all columns together.

```
CREATE TABLE test_two_pk
(
    field1 character varying(50) NOT NULL,
    field2 character varying(50) NOT NULL,
    CONSTRAINT sources_pkey PRIMARY KEY (field1, field2)
);
```

```
CREATE TABLE test_one_pk
```

```
(
    id serial PRIMARY KEY NOT NULL,
    field1 character varying(50) NOT NULL,
    field2 character varying(50) NOT NULL,
    CONSTRAINT sources UNIQUE (field1, field2)
);
```

With these two tables:

in QGIS 2.18, we can add features in both tables.

In QGIS 3.3, we can add features only in `test_one_pk`. It's impossible in `test_two_pk`.

Can we tag this ticket as a regression? As it was working in QGIS 2.

So Nyall, yes I think disabling these checkboxes would be fine. The best would be to check on the QGIS side these multiple primary keys, this is the bonus :-)

#5 - 2018-11-14 08:14 PM - Spencer Gardner

Same issue here. Being able to disable the checkbox would be an acceptable workaround for the time being but I'd strongly prefer a more robust solution. Training other users to catch the issue and mess with checkboxes in the layer setup isn't a sustainable long term solution.

#6 - 2019-02-12 09:09 AM - Regis Haubourg

- Priority changed from Normal to High
- Regression? changed from No to Yes

Hi there, I'm porting QWAT application to QGIS 3 and face this issue again. IMO this is a regression and is related to #19671

From a user point of view, not being able to edit data is a regression - no doubt.

In both cases, QGIS is "over" smart by enforcing DB constraints in the forms.

Here, being able to declare a composite PK would be the solution but this is a new feature. For the #19671, the solution is to bypass the enforcing of a field on QGIS side because the DB has extra triggers that will fill the fields before committing. So we definitely need to be able to disable the constraint constraint checkboxes if needed.

#7 - 2019-03-09 05:04 PM - Giovanni Manghi

- Status changed from Feedback to Open

#8 - 2019-04-07 04:55 AM - Jürgen Fischer

- % Done changed from 0 to 100
- Status changed from Open to Closed

Applied in changeset commit:qgis|2826ba34f9c2a65a419e908d981a7c90447417bc.

Files

Screenshot from 2018-10-10 14-21-31.png	24 KB	2018-10-10	Etienne Trimaille
wrong_default_checkbox.gif	510 KB	2018-10-10	Etienne Trimaille