QGIS Application - Bug report #17185
Data loss when discarding changes in a PostGIS layer

2017-09-22 02:28 PM - Giovanni Manghi

Status:	Closed		
Priority:	High		
Assignee:	Alessandro Pasotti		
Category:	Data Provider/PostGIS		
Affected QGIS version:	2.18.13	**Regression?:**	Yes
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	worksforme
Crashes QGIS or corrupts data:	Yes	**Copied to github as #:**	25084

Description

Import a shape in PostGIS with DB Manager or the QGIS Processing tool

Table is created with PK (not NULL) but not nextval()

Add the table and split one feature

Try save: error because the new feature does not get automatically a new value for the PK

Discard changes > one of the features resulting from the split is gone

Related issues:

Related to QGIS Application - Bug report # 17186: DB Manager and QGIS/Process...	**Closed**	**2017-09-22**
Duplicates QGIS Application - Bug report # 5475: Problem to insert splitted g...	**Closed**	**2012-04-23**

History

#1 - 2017-09-22 02:28 PM - Giovanni Manghi

- Assignee deleted (Jürgen Fischer)

#2 - 2017-09-22 03:07 PM - Jürgen Fischer

Discarding features always implies data loss, doesn't it!?

#3 - 2017-09-22 03:14 PM - Luigi Pirelli

Split a feature is composed of two phases in this order:
1) change geometry of the original geom
2) add new features => error

the rollback remove added feature but not reset old geom because it's change was successfull in the edit buffer => more than a data loss is a not reversible data change.

#4 - 2017-09-22 03:53 PM - Giovanni Manghi

Luigi Pirelli wrote:

> *more than a data loss is a not reversible data change.*

user-side is data loss.

#5 - 2017-09-22 03:55 PM - Giovanni Manghi

Jürgen Fischer wrote:

> *Discarding features always implies data loss, doesn't it!?*

is not a new added geometry that is discarded, but a chunk of an existing geometry (split with the split features tool) that is lost.

#6 - 2017-09-22 03:58 PM - Jürgen Fischer

Giovanni Manghi wrote:

> *Jürgen Fischer wrote:*
>
> > *Discarding features always implies data loss, doesn't it!?*
>
> *is not a new added geometry that is discarded, but a chunk of an existing geometry (split with the split features tool) that is lost.*

It's lost because the user decided to discard it.   The user should fix the feature by assigning a valid key value and proceed with the commit.

#7 - 2017-09-22 04:00 PM - Giovanni Manghi

> *It's lost because the user decided to discarded it.   The user should fix the feature by assigning a valid key value and proceed with the commit.*

you have faith in users that they will click in the message bar and read the log ;) they usually don't, and just run away from the error as fast as they can (by undoing changes).

A fix for #17186 would make this issue much less important.

#8 - 2017-09-22 04:06 PM - Jürgen Fischer

- Related to Bug report #17186: DB Manager and QGIS/Processing import in PostGIS tools are not adding "nextval()" to primary key field added

#9 - 2017-09-22 04:07 PM - Jürgen Fischer

- Duplicates Bug report #7550: Split feature on spatialite layers with primary key error added

#10 - 2017-09-22 04:10 PM - Giovanni Manghi

- Regression? changed from No to Yes

If this was already reported as #7550 and closed as fixed this means that is a regression (I "only" tested down to 2.8).

#11 - 2017-09-22 04:33 PM - Jürgen Fischer

Giovanni Manghi wrote:

> *If this was already reported as #7550 and closed as fixed this means that is a regression (I "only" tested down to 2.8).*

I don't consider it a bug. It's the expected behaviour: You have to specify a key value if you don't use a sequence. And there might by arbitrary other constraints that you have to adjust your attributes (or geometries) to to be able to commit.

#12 - 2017-09-22 04:41 PM - Jürgen Fischer

- Duplicates Bug report #5475: Problem to insert splitted geometries in postgis added

#13 - 2017-09-22 04:41 PM - Jürgen Fischer

- Duplicates deleted (Bug report #7550: Split feature on spatialite layers with primary key error)

#14 - 2017-09-22 04:50 PM - Giovanni Manghi

Jürgen Fischer wrote:

> *Giovanni Manghi wrote:*
>
> > *If this was already reported as #7550 and closed as fixed this means that is a regression (I "only" tested down to 2.8).*
>
> *I don't consider it a bug. It's the expected behaviour: You have to specify a key value if you don't use a sequence. And there might by arbitrary other constraints that you have to adjust your attributes (or geometries) to to be able to commit.*

I think we agree we don't agree :)

my point of view is the following:

I do a whatever editing operation, then **before any save** I discard it. Is not correct that after discarding any unsaved change I have lost data.

#15 - 2017-09-22 04:54 PM - Luigi Pirelli

Jürgen Fischer wrote:

> *Giovanni Manghi wrote:*
>
> > *If this was already reported as #7550 and closed as fixed this means that is a regression (I "only" tested down to 2.8).*
>
> *I don't consider it a bug. It's the expected behaviour: You have to specify a key value if you don't use a sequence. And there might by arbitrary other constraints that you have to adjust your attributes (or geometries) to to be able to commit.*

may we have another API option to setup if to use default nextval or leave it to the user ?

#16 - 2017-09-22 08:19 PM - Jürgen Fischer

Luigi Pirelli wrote:

> *may we have another API option to setup if to use default nextval or leave it to the user ?*

The point is that there is no nextval in this case (see #17186) - otherwise it would be used (merge already doesn't copy attributes with default values; see #5475)

#17 - 2017-09-22 08:52 PM - Jürgen Fischer

Giovanni Manghi wrote:

> *my point of view is the following:*
>
> *I do a whatever editing operation, then **before any save** I discard it. Is not correct that after discarding any unsaved change I have lost data.*

If you would just "discard", it would work like that. But you "commit" - which only partly succeeds and then "discard" the parts that didn't succeed. You wouldn't loose data if you would just update the failed parts and commit them. I agree that partly commits are not ideal and that deducing what actually needs to be done to fix the remaining parts might not be easy. But we still handle adding and deleting, changing of attributes and geometries as separate operations in the provider interface (and hence they run in separate transactions and cannot rolled back cleanly; see QgsVectorLayer::commitChanges).

#18 - 2017-09-24 10:18 AM - Giovanni Manghi

Jürgen Fischer wrote:

> *Giovanni Manghi wrote:*
>
> > *my point of view is the following:*
> >
> > *I do a whatever editing operation, then **before any save** I discard it. Is not correct that after discarding any unsaved change I have lost data.*
>
> *If you would just "discard", it would work like that. But you "commit" - which only partly succeeds and then "discard" the parts that didn't succeed. You wouldn't loose data if you would just update the failed parts and commit them. I agree that partly commits are not ideal and that deducing what actually needs to be done to fix the remaining parts might not be easy. But we still handle adding and deleting, changing of attributes and geometries as separate operations in the provider interface (and hence they run in separate transactions and cannot rolled back cleanly; see QgsVectorLayer::commitChanges).*

ok, thanks I understand better the technicality of what is going on here. But I think that we all agree that from a user point of view this is an unexpected thing. Also is not what we tell people about how editing works (if I'm not wrong in docs we tell people that changes are saved only when hitting the "save" button, regardless of the tool they use).

#19 - 2017-09-24 12:39 PM - Jürgen Fischer

Giovanni Manghi wrote:

> *(if I'm not wrong in docs we tell people that changes are saved only when hitting the "save" button, regardless of the tool they use).*

as said that is what's happening. just that the save partly fails and the failing parts are kept and the successful parts can't be undone.

#20 - 2017-09-24 12:41 PM - Giovanni Manghi

Jürgen Fischer wrote:

> *Giovanni Manghi wrote:*
>
> > *(if I'm not wrong in docs we tell people that changes are saved only when hitting the "save" button, regardless of the tool they use).*
>
> *as said that is what's happening. just that the save partly fails and the failing parts are kept and the successful parts can't be undone.*

so this would be won't fix because impossible to fix? If yes I would add a ticket to remember me to document this behavior.

#21 - 2017-09-24 01:09 PM - Jürgen Fischer

Giovanni Manghi wrote:

> *so this would be won't fix because impossible to fix? If yes I would add a ticket to remember me to document this behavior.*

Of course it's not impossible - but the current behavior is a result of the current provider design (see #17185-17 and #5475-17), so it wouldn't be a small change.

#22 - 2017-09-25 12:13 PM - Giovanni Manghi

> *Of course it's not impossible - but the current behavior is a result of the current provider design (see #17185-17 and #5475-17), so it wouldn't be a small change.*

We should leave this open then. I will add a note to docs to document this. As said, fixing #17186 would makes this slightly less relevant.

#23 - 2017-11-22 01:10 PM - Alessandro Pasotti

- Assignee set to Alessandro Pasotti

#24 - 2017-11-22 01:25 PM - Alessandro Pasotti

- Status changed from Open to Feedback

I canot reproduce the first step in master: importing the table from a shapefile creates the sequence (nextval correctly) also from DB manager

#25 - 2017-11-22 02:30 PM - Alessandro Pasotti

- Status changed from Feedback to Closed

- Resolution set to worksforme

Split features work correctly with a shapefile-imported PG table on both 2.x and master.