

## QGIS Application - Bug report #20682

### QGIS 3.4 ToggleEditing

2018-11-30 11:09 AM - Raul Sangonzalo

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b> Alessandro Pasotti	
<b>Category:</b> Unknown	
<b>Affected QGIS version:</b> 3.5(master)	<b>Regression?:</b> No
<b>Operating System:</b> not relevant	<b>Easy fix?:</b> No
<b>Pull Request or Patch Applied:</b> Yes	<b>Resolution:</b>
<b>Crashes QGIS or corrupts data:</b> No	<b>Copied to github as #:</b> 28502
<b>Description</b>	
<p>We have noticed that the toggleEditing action for vector layers (qgis.utils.iface.actionToggleEditing().trigger()) is causing a problem with our custom/user made feature attribute forms. When the feature attribute form is open and the editing is toggled, the formOpen function of our script runs again. It is as if the toggleEditing action is re-opening the form. This resets any variables we are using to the values set in formOpen. It happens both when the action is called from within the form script or from the built in toolbar edit button. This does not happen in QGIS 3.2, or 2.18. The problem can be easily re-created by calling the toggleEditing action immediately on formOpen - creates a loop. Any help would be greatly appreciated.</p>	
<b>Related issues:</b>	
Related to QGIS Application - Bug report # 19695: Line Edit widgets for integ...	<b>Closed</b> <b>2018-08-25</b>

#### Associated revisions

##### Revision dd309ccb - 2018-11-30 09:14 PM - Alessandro Pasotti

[attributeform] Do not reset feature on editing toggled

... just reset values

Fixes #20682 - form toggle editing

##### Revision c455bf63 - 2018-12-05 09:48 AM - Alessandro Pasotti

[attributeform] Do not reset feature on editing toggled

... just reset values

Fixes #20682 - form toggle editing

cherry-pick dd309ccbe5328165ad725648bf44cc

#### History

##### #1 - 2018-11-30 11:11 AM - Giovanni Manghi

To reach a wider audience you should post your question also on the developers mailing list.

##### #2 - 2018-11-30 11:23 AM - Alessandro Pasotti

- Status changed from Open to Feedback

Raul Sangonzalo wrote:

*This resets any variables we are using to the values set in formOpen.*

I don't understand: if the form was originally opened in readOnly mode (because the layer was not editable) what variable values are you losing?

I believe that the current implementation is correct: the form is basically re-opened because all widgets need to be updated and there are some widgets that have a complete different representation when in edit mode.

### #3 - 2018-11-30 02:53 PM - Raul Sangonzalo

Alessandro Pasotti wrote:

*Raul Sangonzalo wrote:*

*This resets any variables we are using to the values set in formOpen.*

*I don't understand: if the form was originally opened in readOnly mode (because the layer was not editable) what variable values are you losing?*

*I believe that the current implementation is correct: the form is basically re-opened because all widgets need to be updated and there are some widgets that have a complete different representation when in edit mode.*

We set indicator variables in other functions throughout the script, which we use for validation, feature locking, spatial intersection information etc. This is what we lose when we toggleEditing. These variables are usually initially set within the formOpen, and then changed later. So a reset of the form resets these variables. We have found the ability to manage input in the front end (QGIS) to be very valuable. The forms are often left open, with multiple edits and saves occurring before the user closes it.

You are correct that some widgets have a different appearance in edit mode, we have created some of that functionality for our own custom items, by capturing the editingStarted and editingStopped signals (which is a way all widgets could know when to change state). The entire form should not be re-opened until the user chooses to close and re-open it. This is how it has been in previous versions, our forms worked very well, and were stable. This change to the toggleEditing action re-setting the entire form is a very big and unfortunate step backwards for our internal GUI development, as it wipes out many of our control mechanisms at the touch of a single button.

Hopefully this makes sense. Again, any help would be greatly appreciated.

### #4 - 2018-11-30 03:42 PM - Giovanni Manghi

*- Status changed from Feedback to Open*

If confirmed this should be tagged "regression" with "high" priority.

### #5 - 2018-11-30 04:04 PM - Alessandro Pasotti

I'm afraid this is a won't fix, the changes that introduced this issue [\*] were necessary in order to fix other serious issues in the "normal" workflow (for "normal" I mean no custom code or custom forms, just plain QGIS standard forms).

If you've built complex logic inside the formOpen functions you might consider to change your logic into a plugin with custom forms and/or persist the state

of the forms in some other way or instruct your users to not quit the edit state without saving (you can decide to load the custom values only when the layer is editable after all).

I'll have a second look to the code that triggered the change, but I cannot promise anything.

[\*] whether this is an issue or not is debatable: I don't think that it was documented anywhere and it's something that relies on custom code and no public API.

**#6 - 2018-11-30 05:48 PM - Raul Sangonzalo**

Alessandro Pasotti wrote:

*I'm afraid this is a won't fix, the changes that introduced this issue [\*] were necessary in order to fix other serious issues in the "normal" workflow (for "normal" I mean no custom code or custom forms, just plain QGIS standard forms).*

*If you've built complex logic inside the formOpen functions you might consider to change your logic into a plugin with custom forms and/or persist the state of the forms in some other way or instruct your users to not quit the edit state without saving (you can decide to load the custom values only when the layer is editable after all).*

*I'll have a second look to the code that triggered the change, but I cannot promise anything.*

*[\*] whether this is an issue or not is debatable: I don't think that it was documented anywhere and it's something that relies on custom code and no public API.*

While I can't pretend to understand what these other serious issues are/were, I am glad that QGIS3 does seem more stable than QGIS2, and very much want to migrate our system to it.

I would very much appreciate you taking a second look at the code, thank you. And I also appreciate your snippets of advice. It would be a shame if we have to change everything, but we are always up for a challenge!

**#7 - 2018-11-30 09:06 PM - Alessandro Pasotti**

- Status changed from Open to In Progress
- Assignee set to Alessandro Pasotti

**#8 - 2018-11-30 09:07 PM - Alessandro Pasotti**

- Related to Bug report #19695: Line Edit widgets for integer numbers with thousand separators enabled are hard to edit added

**#9 - 2018-11-30 09:15 PM - Alessandro Pasotti**

- Operating System changed from Windows 10 to not relevant
- Pull Request or Patch supplied changed from No to Yes

PR <https://github.com/qgis/QGIS/pull/8581>

**#10 - 2018-11-30 09:16 PM - Alessandro Pasotti**

- Affected QGIS version changed from 3.4.1 to 3.5(master)

**#11 - 2018-12-01 09:42 AM - Anonymous**

- % Done changed from 0 to 100
- Status changed from In Progress to Closed

Applied in changeset commit:qgis|dd309ccbe5328165ad725648bf44ccc0a2e3ecd3.