

QGIS Application - Bug report #4001

addJoinedFeatureAttributes converts join value in substring to string when updating join attributes.

2011-06-19 10:44 PM - Nathan Woodrow

Status:	Closed	
Priority:	Normal	
Assignee:		
Category:	Vectors	
Affected QGIS version:	2.2.0	Regression?: No
Operating System:		Easy fix?: No
Pull Request or Patch supplied:	No	Resolution: fixed/implemented
Crashes QGIS or corrupts data:	No	Copied to github as #: 13989
Description		
<p>If using a join without memoryCache turned on (using Python), the method addJoinedFeatureAttributes converts the join value to a string before passing it to the provider.</p> <pre>qgsvectorlayer.cpp l:1594 subsetString.append("\"" + joinFieldName + "\"" + " = " + "\"" + joinValue.toString() + "\"");</pre> <p>This causes a type mismatch when the provider tries to eval the substring.</p>		

History

#1 - 2014-06-28 07:40 AM - Jürgen Fischer

- Affected QGIS version set to 2.2.0
- Crashes QGIS or corrupts data set to No
- Target version changed from Version 2.0.0 to Future Release - Lower Priority

#2 - 2015-12-03 05:59 AM - Médéric RIBREUX

- Status changed from Open to Feedback
- Pull Request or Patch supplied set to No

Hello, bug triage...

is it still true for QGIS 2.13 (master) ? The code has been heavily changed on this side. I've not made a complete search but I've found this:

<https://github.com/qgis/QGIS/blob/9a89408b72f8bee603a2b3ece66b3905cf6fd04a/src/core/qgsvectorlayerfeatureiterator.cpp#L659> and it seems to be a test for the type of joinValue.

Can we close this bug ?

#3 - 2015-12-27 01:19 PM - Giovanni Manghi

- Resolution set to fixed/implemented
- Status changed from Feedback to Closed

Médéric RIBREUX wrote:

Hello, bug triage...

is it still true for QGIS 2.13 (master) ? The code has been heavily changed on this side. I've not made a complete search but I've found this:
<https://github.com/qgis/QGIS/blob/9a89408b72f8bee603a2b3ece66b3905cf6fd04a/src/core/qgsvectorlayerfeatureiterator.cpp#L659> and it seems to be a test for the type of joinValue.

Can we close this bug ?

closing for lack of feedback and as it seems fixed, please reopen if necessary.