

## QGIS Application - Bug report #20136

### Requesting a feature by its feature id not working with filtered gpkg

2018-10-17 04:16 PM - Matthias Kuhn

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	Even Rouault	
<b>Category:</b>	Data Provider/OGR	
<b>Affected QGIS version:</b>	3.3(master)	<b>Regression?:</b> No
<b>Operating System:</b>		<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b>		<b>Resolution:</b>
<b>Crashes QGIS or corrupts data:</b>		<b>Copied to github as #:</b> 27957
<b>Description</b>		
<p>In the attached dataset, T_Id is a stable primary key, recognized by OGR (ogrinfo says "FID Column = T_Id" and in some debugging of the OGR provider it popped up as fid column name as well).</p> <p>As soon as there is a subset filter string set on the layer, the fid seems to be out of sync.</p> <p>The following code demonstrates the issue. We request feature 1, the returned feature has the id 2:</p> <pre>&gt;&gt;&gt; request = QgsFeatureRequest(1) &gt;&gt;&gt; feature = next(iface.activeLayer().getFeatures(request)) &gt;&gt;&gt; feature.id() 2 &gt;&gt;&gt; feature['T_Id'] 2</pre>		

#### Associated revisions

**Revision 391ec8a5 - 2018-10-18 12:28 PM - Even Rouault**

[OGR provider] Revise significantly the way we handle subset filter to avoid issues with FID (fixes #20136)

Some rationale on this change...

Previously when applying a "regular" subset string, ie. one that is only the content of a where clause, we issued a full "SELECT \* FROM layer WHERE subsetstring", resulting in a OGR SQL layer. The caveat of that is that most OGR drivers will have issues retaining the original FID. A hack consisting in adding a{original\_fid\_name} as orig\_ogc\_fid to the select columns was introduced in <https://github.com/qgis/QGIS/commit/4ce2cf1744b008043403b18b8def8f18c99d14f1> to try to retain the original FID, but this added a lot of complexity. And actually, in the case of the OGR GPKG driver, it caused it to still be confused when analyzing the column definition of the resulting layer, since it sees 2 FID columns despite the renaming (one included in the '\*' wildcard, and the one of orig\_ogc\_fid), which caused it to use sequential FID numbering (the driver when seeing more than once a column that is the FID column assumes that some cross join is done, and thus that FID are unreliable)

A simpler and more robust (crossing fingers!) approach in that case is just to use OGR\_L\_SetAttributeFilter() instead of GDALDatasetExecuteSQL(). Some care must be taken to cancel the filter when removing the subset

filter, or in QgsOgrFeatureIterator when combining with the filter expression coming from the request, but besides that, this is more straightforward, and actually solves #20136

**Revision 763fa420 - 2018-10-19 10:03 AM - Matthias Kuhn**

Merge pull request #8216 from rouault/fix\_20136

[OGR provider] Revise significantly the way we handle subset filter to avoid issues with FID (fixes #20136)

**History**

---

**#1 - 2018-10-17 05:19 PM - Matthias Kuhn**

The filter was set to a simple ``1 = 1``, so it shouldn't filter out anything

**#2 - 2018-10-17 07:24 PM - Even Rouault**

- Assignee set to Even Rouault

**#3 - 2018-10-18 01:17 AM - Even Rouault**

- Status changed from Open to In Progress

<https://github.com/qgis/QGIS/pull/8216>

**#4 - 2018-10-19 10:02 AM - Even Rouault**

- % Done changed from 0 to 100

- Status changed from In Progress to Closed

Applied in changeset commit:qgis|391ec8a5dd4508f75b6538ab7be309379695add4.

**Files**

---

db_curso.gpkg	151 KB	2018-10-17	Matthias Kuhn
---------------	--------	------------	---------------