

QGIS Application - Bug report #19772

QGIS crash when adding feature to Memory Vector Layer

2018-09-05 02:26 PM - Min Min

Status:	Closed	
Priority:	High	
Assignee:		
Category:	Vectors	
Affected QGIS version:	3.0.3	Regression?: No
Operating System:	Windows 10 x64	Easy fix?: No
Pull Request or Patch supplied:	No	Resolution: invalid
Crashes QGIS or corrupts data:	Yes	Copied to github as #: 27597

Description

User Feedback

input files contains 140033 Features of Polygon geometry.

Crash sometime happens, difficult to reproduce. Usually it happens when spanning the map around (drag the mouse around) during loading data into Vector Layer (via Python API).

When trying similar interaction with smaller files (line geomtry with less feature) it doesnt crash.

Some pointers or related issues would be very helpful in troubleshooting !

Report Details

Crash ID: 5525a7236ae259ade34a720056a72353960191f6

Stack Trace

QVector<QVariant>::QVector<QVariant> :
QgsFeature::attributes :
QgsProcessingModelOutput::toVariant :
std::_String_alloc<std::_String_base_types<char,std::allocator<char> > >::_Get_data :
PyCFunction_FastCallDict :
PyObject_GenericGetAttr :
PyEval_EvalFrameDefault :
PyErr_Occurred :
PyBytes_Resize :
PyEval_EvalFrameDefault :
PyFunction_FastCallDict :
PyObject_CallFunctionObjArgs :
PyObject_Call :
PyInit_sip :
QThreadPoolPrivate::reset :
QThread::start :
BaseThreadInitThunk :
RtlUserThreadStart :

QGIS Info

QGIS Version: 3.0.3-Girona
QGIS code revision: commit:8a899c8758
Compiled against Qt: 5.9.2
Running against Qt: 5.9.2
Compiled against GDAL: 2.2.4
Running against GDAL: 2.2.4

System Info

CPU Type: x86_64

History

#1 - 2018-09-05 02:28 PM - Giovanni Manghi

- Status changed from Open to Feedback
- Priority changed from Normal to High
- Crashes QGIS or corrupts data changed from No to Yes

In the meantime please update to the latest version available, and check if is less crashy.

#2 - 2018-09-07 02:25 PM - Min Min

User Feedback

I have tried to reproduce this in later version 3.2.2, and it occurs more frequently. Im doing the following:

I parse multiple geojson files, for each json file, I convert to QgsFeature with geometry, and add those to a memory Vector Layer / data provider. I also check for attributes/fields alignment in Vector layer and in the new data, and update it accordingly (add new attributes/fields into the VectorLayer). Maybe the crash occurs during this process.

Each iteration of adding attributes + feature is done in a ThreadPool of maximum 8 threads. Also after each iteration, I do layer.triggerRepaint() or mapCanvas().refresh() for rendering.

Note: this only happens when I parse large geojson file. In total it consists 140033 features with large MultiPolygon geometries.

Report Details

Crash ID: 0b740b1a81b6b145766026a5ee9136e0c1ec1e6f

Stack Trace

```
QVector<QVariant>::QVector<QVariant> :  
QgsFeature::attributes :  
QgsProcessingModelOutput::toVariant :  
std::_String_alloc<std::_String_base_types<char,std::allocator<char> > >::_Swap_all :  
PyCFunction_FastCallDict :  
PyObject_GenericGetAttr :  
PyEval_EvalFrameDefault :  
PyErr_Occurred :  
PyBytes_Resize :  
PyEval_EvalFrameDefault :  
PyFunction_FastCallDict :  
PyObject_CallFunctionObjArgs :  
PyObject_Call :  
PyInit_sip :  
QThreadPoolPrivate::reset :  
QThread::start :  
BaseThreadInitThunk :  
RtlUserThreadStart :
```

QGIS Info

QGIS Version: 3.2.2-Bonn
QGIS code revision: commit:26842169e9
Compiled against Qt: 5.9.2
Running against Qt: 5.9.2
Compiled against GDAL: 2.2.4
Running against GDAL: 2.2.4

System Info

CPU Type: x86_64
Kernel Type: winnt
Kernel Version: 10.0.15063

#3 - 2018-09-18 08:36 AM - Nyal Dawson

Note: this only happens when I parse large geojson file. In total it consists 140033 features with large MultiPolygon geometries.

This requires a newer version of GDAL where geojson parsing was greatly improved. I'm inclined to close this as an "upstream" issue.

#4 - 2018-09-18 01:34 PM - Min Min

Nyal Dawson wrote:

Note: this only happens when I parse large geojson file. In total it consists 140033 features with large MultiPolygon geometries.

This requires a newer version of GDAL where geojson parsing was greatly improved. I'm inclined to close this as an "upstream" issue.

I am aware of GDAL limitation in parsing large geojson. My workaround is to process geojson file to a Python dict, then parse each single dict entry to QgsFeature. Geometry is parsed using ogr.CreateGeometryFromJson(). This approach of parsing geojson is used in the original bug reports.

The geojson parsing and adding QgsFeature are realized in async manner (using signal/slot mechanism in Qt, AutoConnection). Before, I use nested function or lambda function for signal/slot connection. Then, I switched to use defined function for signal connection, and the error occur much less frequent. So, I am suspecting the error is caused by memory overflow and/or dangling C++ pointer/Python bound method, as sometime during debugging, I got error like QgsFeature does not have signal finished occurring inside the nested function.

Currently, I am rewriting my async code, and hopefully the error will go away. I will keep this thread updated.

#5 - 2018-10-30 10:08 AM - Min Min

I have successfully debugged the cause of this crash and resolved it.

This crash happens when I call provider.addFeatures() in worker thread and the canvas trying to render (via the UI or via layer.triggerUpdate() and canvas.refresh()). Simple solution is to call provider.addFeatures() in main thread.

#6 - 2018-10-30 11:24 AM - Nyal Dawson

- Resolution set to invalid

- Status changed from Feedback to Closed

That's correct - that method is not thread safe and should only be used in the main thread